A lambda-calculus for parallel functions

Gérard Boudol

UNITÉ DE RECHERCHE
INRIA-SOPHIA ANTIPOLIS

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P. 105
78153 Le Chesnay Cedex
France
Tél.: (1) 39 63 55 11

Rapports de Recherche

N° 1231

Programme 1
Programmation, Calcul Symbolique
et Intelligence Artificielle

A LAMBDA-CALCULUS FOR PARALLEL FUNCTIONS

Gérard BOUDOL

Mai 1990

RR-1231

A Lambda-Calculus for Parallel Functions

Un Lambda-Calcul pour les Fonctions Parallèles

Gérard Boudol

INRIA Sophia-Antipolis

06565-VALBONNE FRANCE

Abstract.

We study a λ-calculus enriched with a non-deterministic choice combinator. We show that this combinator is adequately interpreted as the join in a canonical lattice of (parallel) continuous functions. The canonical model is in fact a logical one, concretely presented in a "typing system". Having established the completeness of this system, we show that the canonical interpretation provides us with a fully abstract model of the λ-calculus with join. The proof of completeness relies upon the definability of the finite points of the logical domain.

Résumé.

Nous étudions une extension du λ-calcul, enrichi d'un combinateur de choix non-déterministe. Nous montrons que l'interpretation appropriée de ce combinateur est l'opérateur de borne supérieure dans un treillis canonique de fonctions continues – parallèles. Ce modèle canonique a en fait un contenu logique; ceci est présenté concrètement par un système de "typage". La complétude de ce système étant établie, nous montrons que l'interpretation canonique fournit un modèle complètement abstrait du λ-calcul étendu. La preuve de complétude repose sur la définissabilité des points finis du domaine.

This work has been partly supported by ESPRIT Basic Research Action 3011 CEDISYS.

A Lambda-Calculus for Parallel Functions

Gérard Boudol

INRIA Sophia-Antipolis

06565-VALBONNE FRANCE

Abstract.

We study a λ-calculus enriched with a non-deterministic choice combinator. We show that this combinator is adequately interpreted as the join in a canonical lattice of (parallel) continuous functions. The canonical model is in fact a logical one, concretely presented in a "typing system". Having established the completeness of this system, we show that the canonical interpretation provides us with a fully abstract model of the λ-calculus with join. The proof of completeness relies upon the definability of the finite points of the logical domain.

Content

This work has been partly supported by ESPRIT Basic Research Action 3011 CEDISYS.

1. Introduction.

The main theme of this work is the full-abstraction problem for programming languages. This problem, first raised by Milner (in Milner 73[32]), can be stated as follows: given a programming language, can we provide a mathematical interpretation of this language such that the resulting semantic equality on programs coincides with operational indistinguishability?

According to Scott [40], what is intended by "mathematical interpretation" is an abstract semantics involving objects that do not refer to a particular way of computing. Such an abstract semantics would provide a simple intuition about the meaning of programs. But obviously, programs have to be run on a machine, and an abstract semantics has to be in accordance with operational semantics in some way (see again Scott 70[40], and [8,31]). The full abstraction criterion is one way to assess the accordance between a programming language and a semantics for it. Another naturally arising criterion is the *expressivity*, or *completeness*, of the language with respect to an abstract interpretation: a language is expressive (resp. fully expressive, or universal) if all the finite (resp. computable) objects of the domain of interpretation are definable in the language. As we shall see, these two criteria are in fact closely related.

As a matter of fact, we can turn the full abstraction problem the other way round: given a semantical domain of "objects", can we find a programming language for it, that is a language in which to express these abstract objects? This is the way we shall take: we shall introduce a language which is expressive with respect to a notion of "parallel function". Our notion of parallel function is quite trivial: we shall consider a domain of continuous functions where any function can be regarded as parallel.

Let us recall some well-known results about the full abstraction problem (for a survey see Berry & al. 85[8] and Stoughton 88[45]), in the case of functional languages: this question was investigated by Plotkin [38], who showed that the model of continuous functions over cpo's is not fully abstract for the PCF language (a typed lambda-calculus extended with fixed-points, boolean and arithmetical features). However, this model becomes fully abstract when PCF is enriched with a "parallel or" facility, and the extended language is expressive with respect to the model.

Shortly after, Milner [33] showed that there exists a fully abstract extensional model of PCF. Moreover this model is unique up to isomorphism, and characterized by the fact that it is not "over generous": all its finite elements are definable, i.e. the language is complete (or expressive) with respect to the interpretation. However, Milner's model is built from the syntax and the evaluation mechanism, hence is not "abstract" in the previous intuitive sense. Then we are faced with an alternative: either we restrict the model, trying to find out an abstract notion of sequential function (this is the way explored by Berry and Curien [8,17]), or we extend the language to make it more expressive with respect to the model, as Plotkin did.

We shall look more closely at the situation of the pure λ-calculus with respect to the full abstraction problem, concentrating on Abramsky's ideas [2,3]; for what regards the "classical" λ-calculus we refer to Barendregt's book [6]. Let us first give some general definitions: a programming language consists of a syntax for programs together with an evaluation mechanism, e.g. an abstract machine. This induces a notion of termination which is, according to Abramsky, the only operationally *observable* property, something like "getting the prompt" on the screen. Let $M\Downarrow$ be a notation for "the program M converges" – i.e. M has a terminated evaluation. Conversely divergence is denoted $M\Uparrow$. Then two programs are operationally indistinguishable if by placing them in any program context, we cannot observe any difference. In other words, a program context $C[\,]$ is a *test*, and two programs are operationally equivalent when they pass the same tests, the success

of a test $C[M]$ being its termination. Then operational indistinguishability is given by means of the *testing* or *observation preorder*, whose definition (due to Morris, *cf.* [6] exercise 19.2.5) is as follows:

$$M \sqsubseteq_{\mathcal{O}} N \Leftrightarrow_{\text{def}} \forall C.\, C[M]\Downarrow \Rightarrow C[N]\Downarrow$$

Clearly, this preorder does not tell anything about the abstract nature of the objects involved in the language. An interpretation, assigning a meaning $[\![M]\!]$ to M in some semantic domain (cpo), is said to be

(i) *adequate* if it provides a precongruence, and assigns no significant value to a divergent program, that is:

$$[\![M]\!] \sqsubseteq [\![N]\!] \Rightarrow \forall C.\, [\![C[M]]\!] \sqsubseteq [\![C[N]]\!] \quad \text{and} \quad [\![M]\!] = \bot \Leftrightarrow M\Uparrow$$

(ii) *fully abstract* if it is adequate and reflects the operational distinctions, that is:

$$M \sqsubseteq_{\mathcal{O}} N \Rightarrow [\![M]\!] \sqsubseteq [\![N]\!]$$

Clearly an adequate semantics does not equate operationally distinct programs, that is it satisfies:

$$[\![M]\!] \sqsubseteq [\![N]\!] \Rightarrow M \sqsubseteq_{\mathcal{O}} N$$

For the (pure) λ-calculus, the adequacy requirement can be stated as follows: recall that this calculus is based upon the assertions

$$\begin{aligned}
\beta &: \ (\lambda x M)N = M[N/x] \\
\nu &: \ M = M' \Rightarrow MN = M'N \\
\mu &: \ N = N' \Rightarrow MN = MN' \\
\xi &: \ M = M' \Rightarrow \lambda x M = \lambda x M'
\end{aligned}$$

Then an adequate semantics of the λ-calculus should satisfy these assertions, and also the requirement concerning divergence. This last property depends on the evaluation mechanism. Let us denote by $\vec{\beta}$, $\vec{\nu}$, $\vec{\mu}$ and $\vec{\xi}$ the (conditional) rewriting rules obtained by orienting the preceding equations $M = N$ from left to right. Can these rules be considered as providing an evaluation mechanism?

The answer is as follows: first, there is *no* adequate semantics if we allow all these to be evaluation rules. Let us see this – quite well-known – result in some details: if $\mathbf{K} = \lambda xy.x$ is the usual "right cancellator" (or "left choice"), $\Delta = \lambda x(xx)$ the "duplicator" and $\Omega = \Delta\Delta$ the typically divergent term, and if for any term M we let $F_M = \lambda x((xM)\Omega)$, then the term F_M is divergent, therefore we should have $[\![F_M]\!] = [\![F_N]\!]$ for all M and N, but $F_M\mathbf{K} =_\beta M$, hence $[\![M]\!] = [\![N]\!]$, which is absurd since in an adequate semantics (if any) $[\![\Omega]\!] \neq [\![\mathbf{K}]\!]$. The problem is with $\vec{\mu}$, which has to be dropped (note also that μ is the cause of the difficulties in proving the confluence of evaluation).

Second, if evaluation is given by $\vec{\beta}$, $\vec{\nu}$ and $\vec{\xi}$, then the convergent terms are the ones having a "head normal form"([1]). In this case there is an adequate (i.e. sensible) and fully abstract semantics, namely Scott's D_∞-interpretation studied by Wadsworth in [47] (*cf.* [6] thm. 19.2.9). In particular, Wadsworth showed that "$\beta\nu\xi$-convergence" can be tested within the calculus itself: $M\Downarrow$ if and only if there exists a context C such that $C[M] \xrightarrow{*} \mathbf{I}$ where $\mathbf{I} = \lambda xx$ is the identity (this

([1]) warning: this does not mean that the usual notion of "N is a hnf of M" corresponds to $\beta\nu\xi$-evaluation. For instance if M reduces to N then (xN) "is a hnf of" (xM), but (xM) does not $\beta\nu\xi$-reduce to (xN). What we get are the "principal" hnf.

is Barendregt's notion of solvability). However, the D_∞-interpretation is quite strange, since *no* finite non-trivial element of D_∞ is λ-definable. Here the problem is with the $\vec{\xi}$-rule.

We are then left with $\vec{\beta}$ and $\vec{\nu}$, and the situation appears much better; this is the *lazy* λ-calculus of Abramsky [2]. As a matter of fact, Lévy proposed in [28] an adequate semantics for this calculus, based on "weak head normal forms"; but he abandoned this interpretation because (as he told me) there is no way to test the convergence within the language. In the lazy λ-calculus, with "$\beta\nu$-evaluation", there are many more convergent terms than in the "classical" λ-calculus: closed normal forms are just abstractions $\lambda x M$. In other words, convergent terms are the ones that can accept an input, possibly after some internal evaluation. The typically divergent term is still Ω, but for instance $\lambda x \Omega$, which is clearly "finite", is now strictly greater, in any adequate semantics, than Ω (*cf.* Lévy 76[28], and also the "F-semantics" of Hindley [23,20]). In view of this fact, Abramsky proposed to interpret the lazy λ-calculus in a domain $\boldsymbol{D_\star}$ which is the canonical solution of the equation:

$$D = (D \to D)_\perp$$

where $(X \to Y)$ is the usual domain of continuous functions, extensionally ordered, and $X_\perp$ the lifting construct. So $\boldsymbol{D_\star}$ consists of the domain of all, possibly "non-sequential", continuous functions over itself, plus a non-functional undefined value (then the interpretation will not satisfy the "functionality principle" η, asserting that every object is a function). However, this interpretation is not fully abstract - more precisely, it is over-generous. Two ingredients must be added to the language: *convergence testing* and some (compact) parallel facility (these two can be condensed into what Barendregt calls "parallel or", *cf.* [6] §14.4). Then for the resulting language the $\boldsymbol{D_\star}$-interpretation is fully abstract. Moreover this extended language is expressive: each finite element of $\boldsymbol{D_\star}$ is definable (for all these results, see Abramsky and Ong [2,3,29,30]).

This solves the problem stated above: the λ-calculus with parallel convergence testing provides a language for (finite) parallel functions. However, although we do not give here a precise argumentation about this point, we could see that we only get a limited parallelism in this language: for any term of this calculus, there is a threshold (finite amount of information about the arguments) above which the term is a sequential function.

Here we shall introduce a more expressive language. In our calculus, we separate parallelism from convergence testing, which is represented by the combinator ∇ called the *observer*. Its behaviour will be such that

$$M\Downarrow \;\Leftrightarrow\; (\nabla M) \xrightarrow{*} \mathsf{I}$$

reintroducing a notion of solvability. In the canonical domain $\boldsymbol{D_\star}$ the observer is interpreted as a step function, returning the identity as soon as its argument is non-trivial. As shown by Abramsky and Ong [3], this combinator is needed to test some semantically distinct terms; for instance the observer can be used to separate the terms $\lambda x(x(x(\lambda y\Omega)\Omega)\lambda y\Omega)$ and $\lambda x(x(\lambda z x(\lambda y\Omega)\Omega z)\lambda y\Omega)$.

Let us now turn our attention to parallelism, which seems to be a crucial notion for the full abstraction problem. It is a commonly accepted idea that "full" parallelism (i.e. at the "control" level) entails non-determinism, and then disallows any functional interpretation (*cf.* for instance Milner 73[32], Plotkin 76[37]). A typical non-deterministic operator is the (internal) *choice* $(M \oplus N)$, whose evaluation is given by the rules

$$(M \oplus N) \longrightarrow M \qquad \text{and} \qquad (M \oplus N) \longrightarrow N$$

My claim is that *this operator is exactly what we need to achieve functional parallelism and (together with the observer ∇) full abstraction.* Then the calculus for parallel functions that will

be studied here has the following syntax:

$$M ::= x \mid \lambda x M \mid (MM) \mid (M \oplus M) \mid \nabla$$

To see that full abstraction is achieved, we have to give the abstract meaning of the choice operator. It can be noted that the canonical domain $D_\star$ is in fact an *algebraic (complete) lattice*, with all joins (hence also meets). Then we interpret $\oplus$ as the *join* construction (*sup*, which is typically "non-sequential"):

$$[\![M \oplus N]\!]_\star = [\![M]\!]_\star \sqcup [\![N]\!]_\star$$

Therefore we shall call our calculus the (lazy) λ_J-calculus, that is *λ-calculus with join* (a similar extension of PCF has also been studied by Bloom in [10]).

Introducing the join drastically changes the notion of convergence: a convergent term may now have several normal forms. In particular:

$$(M \oplus N)\Downarrow \;\Leftrightarrow\; M\Downarrow \quad \text{or} \quad N\Downarrow$$

This is the characteristic property of a parallel convergence testing combinator (see Abramsky and Ong [2,3,30] for the definition). To give some intuitive explanation about this calculus, let us say that a terminated computation yields only a partial result – or more accurately: a *part* of the result. If we were able to "join the parts", concurrently evaluated, and perhaps displayed in several "windows"([2]), then we would get the whole result. Technically this means that if $M_1, \ldots, M_k$ are the normal forms of M then we will have, denoting by $\simeq$ the semantic equality:

$$M \simeq M_1 \oplus \cdots \oplus M_k$$

Therefore the non-deterministic choice combinator is better understood as a "parallel fork", initiating concurrent sub-computations. Using this idea of "joining the parts", it is now easy to "program" a parallel disjunction, or a confluent parallel convergence testing, namely:

$$\mathbf{P} =_{\text{def}} (\lambda xy.(\nabla x) \oplus \lambda xy.(\nabla y))$$

Let us see how to define the parallel disjunction: first recall that $\mathbf{K}$ and $\mathbf{F} = \lambda xy.y$ (the "left cancellator", or "right choice") may be regarded as the *truth values*. Then parallel disjunction $\mathbf{O}$ is just the join of "left sequential disjunction" $\mathbf{V}_l = \lambda xy.(x\mathbf{K})y$ and "right sequential disjunction" $\mathbf{V}_r = \lambda xy.(y\mathbf{K})x$ (*cf.* Curien [17]), or more accurately

$$\mathbf{O} =_{\text{def}} \lambda xy.((x\mathbf{K})y \oplus (y\mathbf{K})x)$$

As we shall see, there is another way to understand the semantics of the λ_J-calculus: each term can be viewed as a *stream processor*, accepting strings of values. If we say that a (closed) term M accepts a sequence $R_1 \cdots R_k$ of (closed) terms exactly when $M\,R_1 \cdots R_k\Downarrow$, then $[\![M]\!]_\star \sqsubseteq [\![N]\!]_\star$ if and only if N accepts more strings than M.

As we indicated above, the main result of this paper is that the interpretation of the λ_J-calculus in the canonical domain $D_\star$ provides an adequate and fully abstract semantics for this language. Now let us see how the proof of this result is articulated:

([2]) as we shall see the canonical domain is also a (lower) powerdomain.

(1) the first step is to give a concrete presentation of the canonical domain. This is done using Scott's information systems [43]. Then it can be observed, following the work of Coppo & al., 84 & 83[14,15], that another equivalent concrete presentation is by means of filters of logical formulae. This formalizes Scott's idea that "elements of a domain can be identified with the sets of propositions true of them", which was already realized in Coppo & al., 80 & 83[13,7] (this is also a simple instance of Abramsky's theory of "domains in logical form" [1]). Then we could call $D_\star$ the *logical domain*. The logic is:

$$\phi ::= \omega \mid (\phi \to \phi) \mid (\phi \wedge \phi)$$

together with an axiomatization of the entailment relation $\phi \leq \psi$ meaning intuitively "ϕ implies ψ". This is the logic of Abramsky 88[2], and also, apart from the propositional variables, of Dezani & Margaria 86[20]. Since $[\![M]\!]_\star$ is now a set of formulae, we can recast the interpretation of the λ_J-calculus in this logical presentation of $D_\star$, and we get a "typing system", that is a formal system allowing to prove sequents of the form:

$$x_1 : \phi_1, \ldots, x_n : \phi_n \vdash M : \phi$$

A fact is that, for closed terms:

$$\vdash M : \phi \;\Leftrightarrow\; \phi \in [\![M]\!]_\star$$

Therefore if we define an "assertional preorder", or syntactic logical preorder, by

$$M \sqsubseteq_{\mathcal{S}} N \;\Leftrightarrow_{\text{def}}\; \forall \phi.\ \vdash M : \phi \;\Rightarrow \vdash N : \phi$$

we have

$$[\![M]\!]_\star \sqsubseteq [\![N]\!]_\star \;\Leftrightarrow\; M \sqsubseteq_{\mathcal{S}} N$$

(2) another easy step consists in giving a "realizability interpretation" of the formulae, namely:

$$\begin{aligned} \models M : \omega \;&\Leftrightarrow_{\text{def}}\; \text{true} \\ \models M : (\phi \wedge \psi) \;&\Leftrightarrow_{\text{def}} \models M : \phi \;\&\; \models M : \psi \\ \models M : (\phi \to \psi) \;&\Leftrightarrow_{\text{def}}\; M\Downarrow \;\&\; \forall R.\ \models R : \phi \;\Rightarrow \models MR : \psi \end{aligned}$$

Again associated with this interpretation we find a semantic logical preorder:

$$M \sqsubseteq_{\mathcal{L}} N \;\Leftrightarrow_{\text{def}}\; \forall \phi.\ \models M : \phi \;\Rightarrow \models N : \phi$$

and it is quite easy to show, using an intermediate trace preorder, that this is implied by the testing preorder:

$$M \sqsubseteq_{\mathcal{O}} N \;\Rightarrow\; M \sqsubseteq_{\mathcal{L}} N$$

(3) finally the heart of the full abstraction proof is the *soundness* and *completeness* theorem([3]), that is:

$$x_1 : \phi_1, \ldots, x_n : \phi_n \models M : \phi \;\Leftrightarrow\; x_1 : \phi_1, \ldots, x_n : \phi_n \vdash M : \phi$$

([3]) our completeness result is slightly different from Abramsky's ones, since the notion of realizability we use is purely syntactic.

The key fact for completeness is that any compact element of the domain $D_\star$ is λ_J-definable – in other words, our language is complete with respect to $D_\star$, namely:

$$\forall \phi \; \exists \mathbf{M}_\phi .\; \vdash \mathbf{M}_\phi : \psi \;\Leftrightarrow\; \phi \leq \psi$$

This is the only result for which we need the observer ∇ and the join $\oplus$. The essential rôle of this definability result is not a surprise: although it was not formulated in logical terms in the pioneering work of Plotkin 77[38] and Milner 77[33], the expressiveness of the language was recognized as a crucial property for the full abstraction problem (see also [8], lemma 5.1.2).

Summarizing, we can picture the architecture of the whole proof as follows:

$$\begin{array}{ccc} M \sqsubseteq_{\mathcal{O}} N & \Leftarrow & M \sqsubseteq_\star N \\ \Downarrow & & \Updownarrow \\ M \sqsubseteq_{\mathcal{L}} N & \Leftrightarrow & M \sqsubseteq_{\mathcal{S}} N \end{array}$$

NOTE: although the results are simple adaptations of more or less standard ones, most of the proofs will be given in full details. In fact this work could be regarded as a reformulation of Abramsky's ideas: in particular, the facts concerning the left side of the above diagram are directly adapted from [2,3]. For the lower side, we use proof techniques mainly due to Coppo [7,12,14], Hindley [23,24], Krivine [26]; the idea of the "characteristic terms" $\mathbf{M}_\phi$ is due to Abramsky [2] (see also [16] where Coppo remarked that the proof of completeness relies upon the existence of "characteristic values" for each type). As we shall see the join construct allows us to give a quite simple expression for the $\mathbf{M}_\phi$'s. We shall not give the proofs concerning the right side of the diagram (we refer the interested reader to [15] – but we hope that he/she will think it is an instructive exercise to find out the arguments), mainly because we want to emphasize the "constructive" nature of the canonical interpretation: in fact this interpretation is given by means of the logical sequent calculus.

RELATED WORK. The rest of this introduction is a brief discussion about some semantical ideas, especially concerning non-determinism and concurrency. But let us start with some remarks about domains. Scott originally proposed to work with lattices, and we saw that this seems quite convenient for dealing with "parallel" functions. We should however point out that the intuition we have about the partial order is not exactly the usual one: usually $x \sqsubseteq y$ is interpreted as "x is an incomplete or imperfect entity" (*cf.* Scott 70[40]), possibly arising, in the non-deterministic case (*cf.* [5]), at some stage of computation. Clearly in our case this last interpretation is not correct ([4]): a terminated computation gives only one piece of the value, since reduction is *decreasing* with respect to the semantics, where one loses something each time a choice is made:

$$M \xrightarrow{*} N \;\Rightarrow\; [\![N]\!] \sqsubseteq_\star [\![M]\!]$$

The use of lattices in semantics was criticized, for instance by Plotkin in [39], for the reason that some natural identities – concerning the "if-then-else" construct – fail if the $\top$ element is admited. However, the expected identities refer implicitly to a notion of "type": we cannot guarantee anything if the first argument of the if-then-else is not a "boolean". Clearly in a typed calculus (where typability implies strong normalizability) the semantics of parallel choice $\oplus$ would be quite different from the one proposed here (in fact it would give a meet).

([4]) it could even be the case that "perfect" should mean prime, i.e. join-irreducible.

Another criticism against lattices is presented by Bloom [10] and Meyer [31]: Meyer argued that, although the mathematical results they provide are quite nice, lattices should be rejected because any expressive language for them has to be "unreasonable" – this meaning that the evaluation mechanism has to be inherently non-deterministic (non-confluent). Admittedly, this paper sacrifices determinism to the mathematical results. We shall return to the point of non-determinism later, but let us say a last word about lattices: the meaning of the $\top$ element in Scott's work was not entirely clear. Sometimes, as in [40], $\top$ is an "over-determined", inconsistent element, sometimes it is the most multivalued element, as in [41]. I think both interpretations are right: as the join of all values, $\top$ is clearly the most multivalued element. But then $\top$ does not provide much information: anything is true of it. This is reflected in the syntax: in the "pure" lazy λ-calculus (without the observer and the join) we can define an "ogre", namely:

$$\Xi =_{\text{def}} (\lambda x \lambda y(xx))(\lambda x \lambda y(xx))$$

This combinator is the "best" one, greater than any other. But it is not a very interesting one: since $\Xi \longrightarrow \lambda y \Xi$, the ogre can repeatedly accept any input you propose, but then simply ignores it and returns the prompt.

Now let us say a few words about the full abstraction problem for non-determinism and concurrency. This question, regarding various languages, was investigated by Hennessy in a series of papers ([5,21,22,19,4], see also Darondeau [18]). It should be stressed at once that this work is based on a quite different intuition about non-determinism and parallelism than the one presented here. The difference appears in the treatment of *divergence*, which is the central concept for our semantics. In [22] for instance, and also in Milner 81 [34], a process is said to diverge if it "can compute forever without any communication occurring". Although the operational semantics of the language (namely CCS) is not presented as an evaluation mechanism, we can easily interpret this as a "strong normalizability discipline" – or a "must" convergence, whereas we use here "may" convergence.

Another striking difference concerns the semantical *use* of the notion of divergence: we have adopted a standard "input/output" view, so that a divergent term is meaningless. This is not the case in most works on the semantics of non-determinism and concurrency, where a process is thought of as a "distributed system", made out of several interacting components which can be observed *separately*. Then for instance a compound process $(P \mid \Omega)$, although divergent, is in general regarded as different from Ω since one may communicate with the component P, that is observe it. Obviously to achieve this, the notion of observation has to be more sophisticated than the one used here.

This work evolves from previous attempts by the author to generalize the λ-calculus (there is a growing interest in this area, see [25,35,36,46]). A first one was presented in [11], but it appeared that the constructs were perhaps not the right ones to use, and the choice of weak bisimulation as a semantics certainly was a wrong one. This is reported in [9], where another calculus is proposed. In this last paper the "concurrent abstractions" were reduced to non-deterministic choice. While studying a calculus still involving parallel composition (shuffle), I realized that the appropriate semantics was a functional one. Then, although its presence does not cause particular trouble, parallel composition was left out since its functional interpretation does not seem to be especially appealing. This operator is characterized by the equation:

$$(M \mid N)R \simeq (MR \mid N) \oplus (M \mid NR)$$

2. Syntax and Reduction.

To build the terms of the λ_J-calculus, we assume given an infinite set $\boldsymbol{X}$ of variables. We shall use in what follows various notions of *environment*. For any set E, an E-environment is a mapping $f:\boldsymbol{X} \to E$ which is almost everywhere (i.e. except for finitely many variables) "trivial" – the meaning of this has to be precised in each particular case. The set of E-environments will be denoted $\boldsymbol{Env}(E)$. The *updating* operation on environments is as follows: the updating of f by $e \in E$ at $x \in \boldsymbol{X}$, denoted $[x \mapsto e]f$, is given by

$$([x \mapsto e]f)(y) = \begin{cases} e & \text{if } y = x \\ f(y) & \text{otherwise} \end{cases}$$

2.1 Syntax.

As we just said in the introduction, our calculus is given by the grammar:

$$P ::= x \mid \lambda x P \mid (PP) \mid (P \oplus P) \mid \nabla$$

where x is any variable, $\oplus$ is the *join* (or parallel choice) and ∇ is the *observer* (or convergence testing). We shall use M, N, P, Q, R,... to denote terms, and the set of terms is $\Lambda_J(\nabla)$. The set of "pure" λ-terms, built without $\oplus$ and ∇ is Λ, and $\Lambda(\nabla)$ denotes the set of terms built without join. As usual, we omit some parentheses, denoting (MN) by MN, and use the standard abbreviations, namely $\lambda x_1 \ldots x_n.M$ for $\lambda x_1 \ldots \lambda x_n M$ and $M\, N_1 \cdots N_k$ for $(\cdots(M\, N_1)\cdots N_k)$. The following combinators were already introduced:

identity: $\mathbf{I} = \lambda x x$

duplicator: $\Delta = \lambda x(xx)$

divergence: $\Omega = \Delta\Delta$.

truth values: $\mathbf{T} = \lambda x \lambda y x = \mathbf{K}$ and $\mathbf{F} = \lambda y \lambda x x$.

ogre: $\Xi = (\lambda x \lambda y(xx))(\lambda x \lambda y(xx))$.

The sets of *free* and *bound* variables of a term M, respectively denoted $\mathsf{fv}(M)$ and $\mathsf{bv}(M)$, are defined in an obvious manner. In particular:

$$\mathsf{fv}(M \oplus N) = \mathsf{fv}(M) \cup \mathsf{fv}(N) \quad \text{and} \quad \mathsf{bv}(M \oplus N) = \mathsf{bv}(M) \cup \mathsf{bv}(N)$$

$$\mathsf{fv}(\nabla) = \emptyset = \mathsf{bv}(\nabla)$$

The set of closed terms is $\Lambda_J^\circ(\nabla)$.

For what regards the definition of substitution, we shall follow Stoughton [44]. A *substitution* is an environment $\sigma \in \boldsymbol{Env}(\Lambda_J(\nabla))$, where "trivial" means to take the value Ω (so $\sigma(x) = \Omega$ for almost all x). We use σ, ρ... to range over substitutions. A substitution σ is *closed* if $\sigma(x)$ is a closed term for all x. For V a finite set of variables, the identity substitution ι_V on V, is given by:

$$\iota_V(x) = \begin{cases} x & \text{if } x \in V \\ \Omega & \text{otherwise} \end{cases}$$

To define substitution on terms of $\Lambda_J(\nabla)$, we assume given a choice function, or more precisely a mapping new from finite subsets of $\boldsymbol{X}$ to $\boldsymbol{X}$ such that $\mathsf{new}(V) \notin V$. This assumption makes sense since $\boldsymbol{X}$ is infinite. Then the result $M\sigma$ of applying the substitution σ to the term M is defined by structural induction. We only give the case of abstraction:

$$(\lambda x M)\sigma = \lambda y(N([x \mapsto y]\sigma)) \quad \text{where} \quad y = \mathsf{new}(\{v \mid \exists z \in \mathsf{fv}(M) - \{x\}\ v \in \mathsf{fv}(\sigma(z))\})$$

We use the notation

$$M[N_1/x_1 \cdots N_k/x_k]$$

to denote $M\sigma$ where σ is such that $\sigma(x_i) = N_i$, $\sigma(x) = x$ for $x \in \mathsf{fv}(M) - \{x_1, \ldots, x_n\}$ (and $\sigma(x) = \Omega$ otherwise). The composition of substitutions is given by $(\rho \circ \sigma)(x) = (\sigma(x))\rho$. A trivial adaptation of Stoughton's arguments allows one to prove:

PROPOSITION 2.1.

(i) *if* $\sigma(x) = \rho(x)$ *for any* x *in* $\mathsf{fv}(M)$ *then* $M\sigma = M\rho$

(ii) $(M\sigma)\rho = M(\rho \circ \sigma)$

Then we can define the *syntactic equality* (α-conversion) as in [44], that is:

$$M \equiv N \Leftrightarrow_{\mathrm{def}} \forall \sigma\ M\sigma = N\sigma$$

This relation is a congruence, which can be characterized by:

$$M \equiv N \Leftrightarrow M\iota_V = N\iota_V \quad \text{where} \quad \mathsf{fv}(M) \cup \mathsf{fv}(N) \subseteq V$$

LEMMA 2.2. $\sigma \equiv \rho\ \&\ M \equiv N \Rightarrow M\sigma \equiv N\rho$

In principle, we would have to check the *coherence* of any notion to be introduced, that is: any definition should hold up to syntactic equality. In fact we shall need to show explicitly coherence only for a few notions.

2.2 Reduction and Convergence.

Now we define the *reduction* relation, which will be denoted $M \Rightarrow N$. To this end we need two auxiliary predicates $M \in \boldsymbol{S}$ and $M \in \boldsymbol{N}$ describing the normal forms. They are inductively given by:

(i) $x \in \boldsymbol{S}$ for all $x \in \boldsymbol{X}$

(ii) $M \in \boldsymbol{S} \Rightarrow MN \in \boldsymbol{S}\ \&\ \nabla M \in \boldsymbol{S}$

and

(i) $M \in \boldsymbol{S} \Rightarrow M \in \boldsymbol{N}$

(ii) $\lambda x M \in \boldsymbol{N}$ and $\nabla \in \boldsymbol{N}$

Note that $M \in \boldsymbol{S} \Rightarrow \mathsf{fv}(M) \neq \emptyset$. For instance the terms **I**, **T**, **F** are in $\boldsymbol{N}$.

LEMMA (COHERENCE).

$$M \equiv N \Rightarrow \begin{cases} M \in \boldsymbol{S} \Leftrightarrow N \in \boldsymbol{S} & \text{and} \\ M \in \boldsymbol{N} \Leftrightarrow N \in \boldsymbol{N} \end{cases}$$

PROOF: it is easy to check that $M \in \boldsymbol{S} \Leftrightarrow M\iota_V \in \boldsymbol{S}$ and $M \in \boldsymbol{N} \Leftrightarrow M\iota_V \in \boldsymbol{N}$ if $\mathsf{fv}(M) \subseteq V$ □

The first rules for reduction are the rules of the lazy λ-calculus:

application R1 (β): $(\lambda x M)N \Rightarrow M[N/x]$	R2 (ν): $\dfrac{M \Rightarrow M'}{(MN) \Rightarrow (M'N)}$

The rules for the join are the ones of (internal) non-deterministic choice:

join (left) R3: $(M \oplus N) \Rightarrow M$	*(right)* R4: $(M \oplus N) \Rightarrow N$

Finally the rules concerning the observer state that this combinator waits for the convergence of its argument:

$$\textit{observation R5: } \frac{}{\nabla M \Rightarrow \mathsf{I}} \quad M \in \boldsymbol{N} - \boldsymbol{S} \qquad \text{R6 : } \frac{M \Rightarrow M'}{\nabla M \Rightarrow \nabla M'}$$

We will see the coherence of reduction in a next sub-section. As usual $M \stackrel{*}{\Longrightarrow} M'$ denotes the reflexive and transitive closure of the reduction relation.

DEFINITION (CONVERGENCE).

(i) *M is irreducible, or in normal form, if* $\{M' \mid M \Rightarrow M'\} = \emptyset$

(ii) *M converges, in notation $M\Downarrow$, if it has a normal form, that is there exist an irreducible term N such that* $M \stackrel{*}{\Longrightarrow} N$

(iii) *M is strongly irreducible if MN is irreducible for all N.*

An obvious remark is:

$$M \stackrel{*}{\Longrightarrow} N \ \&\ N\Downarrow \ \Rightarrow \ M\Downarrow$$

The term M is said to diverge, in notation $M\Uparrow$, if it has no normal form. As we shall see (lemma 2.5 below), $M\Downarrow$ means "M is a function", or more accurately "M can accept an input". For instance $\mathsf{I}\Downarrow$ and $\nabla\Downarrow$ (but these two terms are not strongly irreducible), and $\Omega\Uparrow$. Obviously, due to R2, any strongly irreducible term is also irreducible. Note that $(M \oplus N)$ is never irreducible.

We shall show that M is a normal form if and only if $M \in \boldsymbol{N}$. Then reduction in $\Lambda(\nabla)$ is deterministic: there is at most one reduction to perform in a term built without $\oplus$. Therefore in this calculus there is no difference between convergence and strong normalization, that is: $M\Downarrow$ if and only if there is no infinite reduction starting from M. Obviously this is not true for terms of $\Lambda_J(\nabla)$. In particular

$$(M \oplus N)\Downarrow \ \Leftrightarrow \ M\Downarrow \text{ or } N\Downarrow$$

In the next sub-section, which may be skipped at first reading (except perhaps the corollary 2.6 and lemma 2.7), we establish some technical results concerning reduction and convergence.

2.3 Preliminary Results.

LEMMA 2.3. *If $M \Rightarrow N$ then for any substitution σ $\exists P \equiv N\sigma \quad M\sigma \Rightarrow P$.*

PROOF: by induction on the proof of the transition. If the reduction $M \Rightarrow M'$ is proved by R1, we have $M = (\lambda x P)N$ and $M' = P[N/x]$. By definition of substitution $M\sigma = (\lambda y(P[x \mapsto y]\sigma))N\sigma$ where $y = \mathsf{new}(\{v \mid \exists z \in \mathsf{fv}(M)\ v \in \mathsf{fv}(\sigma(z))\})$. Then by R1

$$M\sigma \Rightarrow (P([x \mapsto y]\sigma))[N\sigma/y]$$

By the proposition 2.1 (ii) we have

$$(P([x \mapsto y]\sigma))[N\sigma/y] = P([y \mapsto N\sigma]\iota) \circ ([x \mapsto y]\sigma)$$

Since $y \neq x \ \Rightarrow \ y \notin \{v \mid \exists z \in \mathsf{fv}(M)\ v \in \mathsf{fv}(\sigma(z))\})$, we have by the proposition 2.1 (i):

$$(P([x \mapsto y]\sigma))[N\sigma/y] = P([x \mapsto N\sigma](\iota \circ \sigma))$$

Moreover

$$[x \mapsto N\sigma](\iota \circ \sigma) \equiv [x \mapsto N\sigma]\sigma = \sigma \circ ([x \mapsto N]\iota)$$

therefore by the lemma 2.1 and proposition 2.1 (ii)

$$(P([x \mapsto y]\sigma))[N\sigma/y] \equiv (P[N/x])\sigma = M'\sigma$$

All the other cases are trivial (for R5 we just have to note that if $M \in \boldsymbol{N} - \boldsymbol{S}$ then $M\sigma \in \boldsymbol{N} - \boldsymbol{S}$ for any substitution σ) □

LEMMA (COHERENCE).

(i) $M \Rightarrow N \;\&\; M' \equiv M \;\Rightarrow\; \exists N' \equiv N \quad M' \Rightarrow N'$

(ii) $M \equiv N \;\Rightarrow\; M{\Downarrow} \;\Leftrightarrow\; N{\Downarrow}$

PROOF: an easy consequence of the previous lemma is

$$M \Rightarrow N \;\&\; \mathsf{fv}(M) \subseteq V \;\Rightarrow\; \exists N' \equiv N\iota_V \quad M\iota_V \Rightarrow N'$$

Then to prove the first point it is enough to show

$$M\iota_V \Rightarrow N \;\Rightarrow\; \exists N' \equiv N \quad M \Rightarrow N'$$

The straightforward proof, by induction on the inference of the transition, is omitted. The second point is an easy consequence of the previous one □

An obvious consequence of these two lemma is:

COROLLARY 2.4. *If* $M \overset{*}{\Longrightarrow} N$ *and* $P \equiv M\sigma$ *then* $\exists Q \equiv N\sigma \quad P \overset{*}{\Longrightarrow} Q$.

LEMMA 2.5.

(i) *M strongly irreducible* $\Leftrightarrow$ $M \in \boldsymbol{S}$

(ii) *M irreducible* $\Leftrightarrow$ $M \in \boldsymbol{N}$.

PROOF: the implications

$$M \in \boldsymbol{S} \;\Rightarrow\; M \text{ strongly irreducible}$$
$$M \in \boldsymbol{N} \;\Rightarrow\; M \text{ irreducible}$$

are easily proved by induction on the definition of the predicates $M \in \boldsymbol{S}$ and $M \in \boldsymbol{N}$.

Conversely, we show

$$M \notin \boldsymbol{S} \;\Rightarrow\; M \text{ not strongly irreducible}$$
$$M \notin \boldsymbol{N} \;\Rightarrow\; M \text{ not irreducible}$$

by induction on the term M. Assume first that $M \notin \boldsymbol{S}$. Then

• M cannot be a variable. If M is ∇ or an abstraction $\lambda x N$, then using R5 (namely $\nabla(\lambda x N) \Rightarrow \mathbf{I}$) or R1 we see that M is not strongly irreducible.

• if $M = (PQ)$ then $P \notin \boldsymbol{S}$ (and $P \neq \nabla$) or $P = \nabla$ and $Q \notin \boldsymbol{S}$. In the first case, by induction hypothesis $PR \Rightarrow P'$ for some R and P', and it is easy to show, by case on the proof of this reduction (which can only be proved using R1 or R2) that $M \Rightarrow M'$ for some M'. In the second case, we have either $Q \in \boldsymbol{N}$, and R5 shows that M is not irreducible (hence not strongly irreducible), or $Q \notin \boldsymbol{N}$, and by induction hypothesis Q is not irreducible. Then we use R6 to conclude. Note that this proves $(PQ) \notin \boldsymbol{S} \;\Rightarrow\; \exists M' \; (PQ) \Rightarrow M'$.

• it is easy to see that if $M = (P \oplus Q)$ then M is not strongly irreducible (using R3 or R4 and R2).

Assume now $M \notin \boldsymbol{N}$. Then

• M cannot be a variable, nor the observer ∇ or an abstraction $\lambda x N$. If $M = (P \oplus Q)$ then R3 or R4 show that M is not irreducible.

• if $M = (PQ)$ then $M \notin \boldsymbol{S}$, and the argument above shows that there exists M' such that $M \Rightarrow M'$ □

We can now prove that the rôle of the observer is to detect the convergence of its argument – and to return the identity in this case:

COROLLARY 2.6. *For any closed term M*

$$M\Downarrow \Leftrightarrow (\nabla M) \stackrel{*}{\Longrightarrow} \mathbf{I} \Leftrightarrow (\nabla M)\Downarrow$$

PROOF: if $M\Downarrow$ then, by the previous lemma, $\exists N \in \boldsymbol{N}\ M \stackrel{*}{\Longrightarrow} N$. Since M is closed, this is also true for N, therefore $N \in \boldsymbol{N} - \boldsymbol{S}$ (for $N \in \boldsymbol{S} \Rightarrow \mathsf{fv}(N) \neq \emptyset$). An easy induction on the length of the reduction $M \stackrel{*}{\Longrightarrow} N$ shows that $\nabla M \Rightarrow \mathbf{I}$ (using possibly R6, and R5), and $(\nabla M)\Downarrow$ since $\mathbf{I}\Downarrow$.

Conversely, one first observe that if M is closed, then ∇M is not irreducible (since either $M \in \boldsymbol{N} - \boldsymbol{S}$, in which case we can apply R5, or $M \notin \boldsymbol{N}$, in which case the previous lemma shows that we can apply R6). Then it is easily seen, by induction on the inference of the reductions, that if $\nabla M\Downarrow$ then $\nabla M \stackrel{*}{\Longrightarrow} \mathbf{I}$, and that this implies $\exists M' \in \boldsymbol{N} - \boldsymbol{S}.\ M \stackrel{*}{\Longrightarrow} M'$, that is $M\Downarrow$ □

It should be clear that if the operator M in an application MN is divergent, then MN is divergent: we have to evaluate the operator in an operator/operand combination. More precisely, we have to find first a normal form for the operator, that is:

LEMMA 2.7. $MR \stackrel{*}{\Longrightarrow} N \in \boldsymbol{N} \Leftrightarrow \exists Q \in \boldsymbol{N}.\ M \stackrel{*}{\Longrightarrow} Q\ \&\ QR \stackrel{*}{\Longrightarrow} N$

PROOF: if $M \stackrel{*}{\Longrightarrow} Q$ we have, using R2, $MR \stackrel{*}{\Longrightarrow} QR$, therefore

$$M \stackrel{*}{\Longrightarrow} Q\ \&\ QR \stackrel{*}{\Longrightarrow} N \Rightarrow MR \stackrel{*}{\Longrightarrow} N$$

We prove the converse part of the assertion by induction on the length of the reduction $MR \stackrel{*}{\Longrightarrow} N$. This is obvious if this length is 0, since $MR \in \boldsymbol{N} \Rightarrow M \in \boldsymbol{N}$ (using the lemma 2.5). Assuming that there exists P such that $MR \Rightarrow P \stackrel{*}{\Longrightarrow} N$, we proceed by induction on the proof of the reduction $MR \Rightarrow P$:

• if the rule used to prove this reduction is R1 then we can let $Q = M$ since $M = \lambda x M'$, hence $M \in \boldsymbol{N}$.

• if the last rule used to prove this reduction is R2 then $P = M'R$ with $M \Rightarrow M'$, and we use the induction hypothesis on the length of the reduction to conclude.

• if $MR \Rightarrow P$ is proved using R5 or R6, we have $M = \nabla$, hence we can let $Q = M$ □

For instance if we define $T = \lambda x \nabla x x$ we will have:

$$M\Downarrow \Leftrightarrow TM \stackrel{*}{\Longrightarrow} M$$

3. Semantics.

In this section we introduce some preorders on terms, wich will be denoted $\sqsubseteq$, decorated with various subscripts. The associated equivalences will be denoted $\simeq$ (with the corresponding subscripts). In fact there is another hidden parameter on which all our preorders depend, namely the language on which they are defined: most of the results below hold for any extension $\Lambda(S)$ of the λ-calculus, where S is a set of λ_J-combinators, that is closed terms of $\Lambda_J(\nabla)$.

We first define a general notion of *semantics* for our language, which is a preorder compatible with the syntactic constructs, satisfying some other requirements. To state the precongruence property, we use the standard notion of *context*: a context is a term built using the language constructs, and possibly a new constant $[]$, the *hole*. Filling the hole with the context B in the context C gives the context $C[B]$ (the definition, by structural induction, is obvious). Note that in this operation, some free variables of B may be bound by the context C, e.g. $(\lambda x[])[M] = \lambda x M$. We shall say that C *closes* B if $C[B]$ is closed.

The definition of most of our preorders relies upon an *operational criterion* according to which a convergent term is better than any other one, that is:

$$M \prec N \;\Leftrightarrow_{\text{def}}\; M{\Downarrow} \;\Rightarrow\; N{\Downarrow}$$

This is a (coherent) preorder, which is not a precongruence: for instance $\mathbf{F} \prec \mathbf{I}$ but $\mathbf{F}\Omega \not\prec \mathbf{I}\Omega$. Note that

$$M \stackrel{*}{\Longrightarrow} N \;\Rightarrow\; N \prec M$$

A preorder on terms is a semantics if it satisfies the following:

DEFINITION (SEMANTICS) 3.1. *A preorder $\sqsubseteq$ (whose associated equivalence is $\simeq$) on terms is a semantics if it is*

(i) *coherent:* $M \equiv N \;\Rightarrow\; M \sqsubseteq N$

(ii) a *precongruence:* $M \sqsubseteq N \;\Rightarrow\; \forall C.\, C[M] \sqsubseteq C[N]$

(iii) a *model of β:* $(\lambda x M)N \simeq M[N/x]$

(iv) *computationally adequate:* $M \sqsubseteq N \;\Rightarrow\; M \prec N$ *for closed terms M and N.*

Note that the laws μ and ξ are valid in any semantics, that is:

$$(\mu)\quad M \simeq M' \;\Rightarrow\; NM \simeq NM'$$
$$(\xi)\quad M \simeq M' \;\Rightarrow\; \lambda x M \simeq \lambda x M'$$

Therefore these laws may be used to perform some simplifications or optimizations for instance.

3.1 Testing.

Our first preorder is the *testing* preorder, already mentioned in the introduction. As we said, the *tests* are simply contexts, and the success of a test $C[\,]$ applied to M is the convergence of $C[M]$. In the definition, we require the tested terms to be closed by the context:

DEFINITION (TESTING or OBSERVATION PREORDER).

$$M \sqsubseteq_{\mathcal{O}} N \;\Leftrightarrow_{\text{def}}\; \forall C \text{ closing } M \;\&\; N.\; C[M] \prec C[N]$$

This is a slight variation on Abramsky's definition [2]: we do not require the terms M and N to be closed. Therefore it is obvious that this preorder is a precongruence, for $C[C'[M]] = (C[C'])[M]$. Clearly any semantics is a refinement of the testing preorder:

$$M \sqsubseteq N \;\Rightarrow\; M \sqsubseteq_{\mathcal{O}} N$$

A semantics $\sqsubseteq$ is fully abstract if the converse is also true:

DEFINITION (FULL ABSTRACTION). *A semantics $\sqsubseteq$ is fully abstract if it satisfies:*

$$M \sqsubseteq_{\mathcal{O}} N \;\Rightarrow\; M \sqsubseteq N$$

In the proof of our full abstraction result, we shall see that we can assume the testing ability to be restricted to *applicative tests*, namely:

$$A ::= [] \mid \lambda x A \mid (AM)$$

where M is any term. The corresponding testing preorder, that is *applicative testing*, is denoted $\sqsubseteq_{\mathcal{A}}$. For instance $\mathbf{T} \not\simeq_{\mathcal{A}} \mathbf{F}$, since these two terms are distinguished by the test $[]\Omega\mathbf{I}$. Similarly the terms $\mathbf{I}$ and $\mathbf{A} = \lambda x(\lambda y(xy))$ are distinguished by $[]\Omega$.

In the rest of this section we give some alternative characterizations of applicative testing, as a trace preorder, and by means of a kind of "logical relation". More precisely, we prove that applicative testing corresponds to an extensional preorder on terms. We cannot define extensional equality simply as the least relation $\simeq$ such that (for closed terms):

$$M \simeq N \;\Leftrightarrow\; \forall R.\, MR \simeq NR$$

since this relation is inconsistent ($M \simeq N$ for all M and N is the least fixed-point of this recursive definition). We have to incorporate a type distinction, giving some information on the functional character of M (and N). In fact this distinction is given by our operational criterion: a (closed) term is "functional" if it is convergent. The definition of the extensional preorder, due to Abramsky [2] and inspired by the well-known Park & Milner's notion of bisimulation, is as follows:

DEFINITION and LEMMA (APPLICATIVE SIMULATION and EXTENSIONAL PREORDER).
A relation $\mathcal{R}$ on closed term is an applicative simulation if

$$M \,\mathcal{R}\, N \;\Rightarrow\; \begin{cases} M \prec N & \text{and} \\ \forall Q \text{ closed}.\ (MQ)\,\mathcal{R}\,(NQ) \end{cases}$$

The relation on closed terms given by

$$M \sqsubseteq^{\circ}_{\mathcal{E}} N \;\Leftrightarrow_{\text{def}}\; \exists \mathcal{R} \quad \textit{applicative simulation} \quad M\mathcal{R}N$$

is an applicative simulation and a preorder, called the extensional preorder.

(one shows that the composition of two applicative simulations is an applicative simulation). The extensional preorder is extended to arbitrary terms by instanciation:

$$M \sqsubseteq_{\mathcal{E}} N \;\Leftrightarrow_{\text{def}}\; \forall \sigma \text{ closed } M\sigma \sqsubseteq^{\circ}_{\mathcal{E}} N\sigma$$

As we shall see, the extensional preorder coincide with a trace preorder, which we define now:

DEFINITION (TRACE PREORDER). *The set $\mathcal{T}(M)$ of traces of a closed term M is the set of (possibly empty) sequences $R_1 \cdots R_k$ of closed terms such that $M\, R_1 \cdots R_k \Downarrow$. The trace preorder on closed terms is given by*

$$M \sqsubseteq^{\circ}_{\mathcal{T}} N \;\Leftrightarrow_{\text{def}}\; \mathcal{T}(M) \subseteq \mathcal{T}(N)$$

This preorder is extended to arbitrary terms by instanciation:

$$M \sqsubseteq_{\mathcal{T}} N \;\Leftrightarrow_{\text{def}}\; \forall \sigma \text{ closed } M\sigma \sqsubseteq^{\circ}_{\mathcal{T}} N\sigma$$

Let us see some properties of this preorder. We use $U, V \ldots$ to denote traces, UV their concatenation and ε the empty trace. Using the lemma 2.7 it is easy to see that

REMARK 3.2. $U \in \mathcal{T}(M) \;\Leftrightarrow\; \exists N \in \boldsymbol{N}.\, M \stackrel{*}{\Longrightarrow} N \;\&\; U \in \mathcal{T}(N)$

Then we have, for closed terms:

$$M{\Downarrow} \;\Leftrightarrow\; \mathcal{T}(M) \neq \emptyset \;\Leftrightarrow\; \varepsilon \in \mathcal{T}(M)$$

An obvious consequence is the following:

$$M{\Uparrow} \;\Rightarrow\; \forall N.\, M \sqsubseteq^{\circ}_{\mathcal{T}} N$$

For example Ω is a least element for this preorder. Similarly, if N is a closed terms such that

$$\forall k\; \forall R_1, \ldots, R_k.\; N\, R_1 \cdots R_k{\Downarrow}$$

then $M \sqsubseteq^{\circ}_{\mathcal{T}} N$ for any M. For instance $M \sqsubseteq^{\circ}_{\mathcal{E}} \Xi$ for all closed terms M, and similarly $M \sqsubseteq^{\circ}_{\mathcal{E}} \Theta\mathbf{K}$ where $\Theta = (\lambda xy.(y(xx)y))(\lambda xy.(y(xx)y))$ is the Turing's fixed-point combinator.

We can give the trace interpretation of the constructions of the λ_{J}-calculus (for closed terms), as follows:

$$\begin{aligned}
\mathcal{T}(MN) &= \{U \mid NU \in \mathcal{T}(M)\} \\
\mathcal{T}(\lambda x M) &= \{\varepsilon\} \cup \{NU \mid U \in \mathcal{T}(M[N/x])\} \\
\mathcal{T}(M \oplus N) &= \mathcal{T}(M) \cup \mathcal{T}(N) \\
\mathcal{T}(\nabla) &= \begin{cases} \{MU \mid U \in \mathcal{T}(\mathbf{I})\} & \text{if } \mathcal{T}(M) \neq \emptyset \\ \emptyset & \text{otherwise} \end{cases}
\end{aligned}$$

It is easy to see that, for closed terms:

REMARK (VALIDITY of β) 3.3. $(\lambda x M)N \simeq_{\mathcal{T}} M[N/x]$

For instance if $x \notin \mathsf{fv}(M)$ we have $(\lambda x(Mx))N \simeq_{\mathcal{T}} MN$, and since $M \prec \lambda x(Mx)$ we get (*cf.* Scott 80[42])

$$\eta^{-} :\quad M \sqsubseteq_{\mathcal{T}} \lambda x(Mx) \qquad (x \text{ not free in } M)$$

Moreover one can check that

$$M{\Downarrow} \;\Rightarrow\; M \simeq_{\mathcal{T}} \lambda x(Mx) \qquad (x \text{ not free in } M)$$

Now we show that the trace preorder and extensional preorder coincide:

LEMMA 3.4. $M \sqsubseteq_{\mathcal{E}} N \;\Leftrightarrow\; M \sqsubseteq_{\mathcal{T}} N$

PROOF: clearly the relation $\sqsubseteq^{\circ}_{\mathcal{T}}$ is an applicative simulation, hence $M \sqsubseteq_{\mathcal{T}} N \;\Rightarrow\; M \sqsubseteq_{\mathcal{E}} N$. Conversely, one easily proves by induction on k that if $\mathcal{R}$ is an applicative simulation then

$$M\,\mathcal{R}\,N \;\Rightarrow\; \forall R_1, \ldots, R_k.\, (M\, R_1 \cdots R_k)\,\mathcal{R}\,(N\, R_1 \cdots R_k)$$

Moreover $P\,\mathcal{R}\,Q \;\Rightarrow\; P \prec Q$ by definition, hence the lemma ❑

Using this result and the remark 3.2, one can see that:

LEMMA (REDUCTION is DECREASING) 3.5. $M \stackrel{*}{\Longrightarrow} N \;\Rightarrow\; N \sqsubseteq_{\mathcal{E}} M$

Another consequence of the lemma 3.4 is that $\sqsubseteq_{\mathcal{E}}^{\circ}$ is the least fixed-point of the definition of applicative simulations. Then to show that $M \sqsubseteq_{\mathcal{E}} N$ it is enough to show that for all closed substitution σ we have $M\sigma \prec N\sigma$ and $(M\sigma)R \sqsubseteq_{\mathcal{E}} (N\sigma)R$ for all closed terms R. For instance, we can prove using this argument that the extensional preorder is compatible with applicative contexts:

LEMMA 3.6. $M \sqsubseteq_{\mathcal{E}} N \Rightarrow \forall A.\ A[M] \sqsubseteq_{\mathcal{E}} A[N]$

PROOF: by induction on the applicative context A, we show that for any closed substitution σ and closed term R: $A[M]\sigma \prec A[N]\sigma$ and $(A[M]\sigma)R \sqsubseteq_{\mathcal{E}} (A[N]\sigma)R$. This is trivial for $A = []$.

We have $(\lambda x A[M])\sigma = \lambda y(A[M]([x \mapsto y]\sigma))$ for some new variable y. Then $(\lambda x A[M])\sigma\Downarrow$, and similarly $(\lambda x A[N])\sigma\Downarrow$, therefore $(\lambda x A[M])\sigma \prec (\lambda x A[N])\sigma$. Let R be a closed term; then

$$\begin{aligned}
((\lambda x A[M])\sigma)R &= (\lambda y A[M]([x \mapsto y]\sigma))R \\
&\sqsubseteq_{\mathcal{E}} (A[M]([x \mapsto y]\sigma))[R/y] && \text{(validity of } \beta) \\
&= A[M]([x \mapsto R]\sigma) && \text{(proposition 2.1)} \\
&\sqsubseteq_{\mathcal{E}} A[N]([x \mapsto R]\sigma) && \text{(induction hypothesis)} \\
&= (A[N]([x \mapsto z]\sigma))[R/z] \\
&\sqsubseteq_{\mathcal{E}} (\lambda z A[N]([x \mapsto z]\sigma))R && \text{(validity of } \beta) \\
&= ((\lambda x A[N])\sigma)R
\end{aligned}$$

Finally for a test of the form AP we have $((AP)[M])\sigma = ((A[M]P)\sigma = (A[M]\sigma)(P\sigma)$. By induction hypothesis $A[M]\sigma \sqsubseteq_{\mathcal{E}} A[N]\sigma$, therefore $(A[M]\sigma)(P\sigma) \sqsubseteq_{\mathcal{E}} (A[N]\sigma)(P\sigma)$ since $\sqsubseteq_{\mathcal{E}}$ is an applicative simulation, hence $(AP)[M]\sigma \prec (AP)[N]\sigma$ and for any closed term R we have

$$(A[M]\sigma)(P\sigma)R \sqsubseteq_{\mathcal{E}} (A[N]\sigma)(P\sigma)R$$

that is $(AP)[M]\sigma \sqsubseteq_{\mathcal{E}} ((AP[N])\sigma$ □

An obvious consequence is:

COROLLARY 3.7. $M \sqsubseteq_{\mathcal{E}} N \Rightarrow M \sqsubseteq_{\mathcal{A}} N$

The next result indicates that the converse implication holds, that is the extensional preorder is weaker than applicative testing:

LEMMA 3.8. $M \sqsubseteq_{\mathcal{A}} N \Rightarrow M \sqsubseteq_{\mathcal{E}} N$

PROOF: we show that the relation $\mathcal{R}$ consisting of the pairs $(M\sigma, N\sigma)$ where $M \sqsubseteq_{\mathcal{A}} N$ and σ is closed is an applicative simulation. Let $R_i = \sigma(x_i)$ where $\mathsf{fv}(M) \cup \mathsf{fv}(N) = \{x_1, \ldots, x_n\}$. It is easy to see that, since the R_i's are closed:

$$M[R_1/x_1 \cdots R_n/x_n] \equiv M[R_1/x_1] \cdots [R_n/x_n]$$

Then if A is the applicative test $(\lambda x_1 \ldots x_n.[])R_1 \cdots R_n$ (which closes both M and N), we can see that $M\sigma\Downarrow \Leftrightarrow A[M]\Downarrow$, and similarly for N. Therefore $M\sigma \prec N\sigma$ since $A[M] \prec A[N]$.

Now let Q be a closed term; since obviously $M \sqsubseteq_{\mathcal{A}} N \Rightarrow MQ \sqsubseteq_{\mathcal{A}} NQ$ and $(MQ)\sigma \equiv (M\sigma)Q$, we have

$$M \sqsubseteq_{\mathcal{A}} N \Rightarrow (M\sigma)Q\ \mathcal{R}\ (N\sigma)Q$$

that is the second clause of the definition of applicative simulations □

Summarizing, we have established:

$$M \sqsubseteq_{\mathcal{O}} N \Rightarrow M \sqsubseteq_{\mathcal{A}} N \Leftrightarrow M \sqsubseteq_{\mathcal{E}} N \Leftrightarrow M \sqsubseteq_{\mathcal{T}} N$$

3.2 The Canonical Domain.

In this section we introduce the canonical interpretation of the λ_J-calculus. However, we must point out that this section is quite allusive – in particular no proof will be given. The reason is that we consider as the actual interpretation the *logical interpretation*, given in the next section. Here we will begin with some rather abstract concepts, and introduce the logical content of these concepts by gradual changes in the notations.

Let us recall that, following Scott's work (see Barendregt [6]), we can interpret the λ-calculus in any reflexive domain, that is a domain D such that the domain $(D \to D)$ of continuous functions over D is a retract of D. More specifically, we shall interpret the λ_J-calculus in Abramsky's domain, satisfying the equation:

$$D_\star = (D_\star \to D_\star)_\perp$$

where $D_\perp$ is the usual *lifting* of D. There are two obvious mappings associated with the lifting construct:

$$\mathsf{down}: D_\star \to (D_\star \to D_\star) \qquad \text{and} \qquad \mathsf{up}: (D_\star \to D_\star) \to D_\star$$

satisfying $\mathsf{down}(\mathsf{up}(f)) = f$. Then we can interpret the λ-calculus as usual: $[\![M]\!]$ is a mapping from environments $\rho \in \mathit{Env}(D_\star)$ (where "trivial" means undefined, that is $\rho(x) = \perp$ for almost all x) to $D_\star$ given by

$$[\![x]\!]\rho = \rho(x)$$
$$[\![MN]\!]\rho = (\mathsf{down}([\![M]\!]\rho))([\![N]\!]\rho)$$
$$([\![\lambda x M]\!]\rho)e = \mathsf{up}(f_{x,\rho}) \quad \text{where } f_{x,\rho}(e) = [\![M]\!]([x \mapsto e]\rho)$$

The observer ∇ is interpreted as a step function, where $\mathrm{I}_\star$ is the identity function on $D_\star$:

$$[\![\nabla]\!](d) = \begin{cases} \mathrm{I}_\star & \text{if } d \neq \perp \\ \perp & \text{otherwise} \end{cases}$$

The domain $D_\star$ is in fact a lattice, so that we can define:

$$[\![M \oplus N]\!]\rho = ([\![M]\!]\rho) \sqcup ([\![N]\!]\rho)$$

A standard fact is that $[\![M]\!]$ is a continuous function from $\mathit{Env}(D_\star)$ (which is a "weak product", ordered componentwise) to $D_\star$. Then we can define an extensional semantic preorder on terms by

$$M \sqsubseteq_\star N \Leftrightarrow_{\mathrm{def}} \forall \rho.\ [\![M]\!]\rho \sqsubseteq [\![N]\!]\rho$$

In the next sub-sections we will give an explicit description of the canonical domain $D_\star$, using Scott's information systems [43]. However, we do not need the concept of information system in its full generality, since we only deal with lattices.

3.2.1 Algebraic Lattices and Information Systems.

We first recall some standard definitions concerning lattices. A *complete lattice* is a partial order $(L, \sqsubseteq)$ such that each subset X of L has a least upper bound (join) $\bigsqcup X$ (hence also a greatest lower bound). A complete lattice has a least and a greatest element, namely $\perp = \bigsqcup \emptyset$ and $\top = \bigsqcup L$. The

lifting construct builds a complete lattice $L_\perp$ out of a given complete lattice L: this simply consists in adding a new bottom element. A subset X of L is *directed* if for any finite subset Y of X there exists $x \in X$ such that $\bigsqcup Y \sqsubseteq x$ (therefore a directed set is non-empty: $\perp \in X$). Given two complete lattices L_0 and L_1, a function $f: L_0 \to L_1$ is *continuous* if it preserves the joins of directed subsets, that is if for any directed subset X of L we have $f(\bigsqcup X) = \bigsqcup f(X)$. The set $(L_0 \to L_1)$ of continous functions, extensionally ordered (i.e. $f \sqsubseteq g \Leftrightarrow_{\text{def}} \forall x\ f(x) \sqsubseteq_1 g(x)$) is a complete lattice.

An element c of a complete lattice L is *compact* iff for any directed subset X of L such that $c \sqsubseteq \bigsqcup X$ there exists $x \in X$ such that $c \sqsubseteq x$. We shall use $\mathcal{K}(L)$ to denote the set of compact elements of L. A complete lattice is *algebraic* if any element e of L is the join of the compact points it dominates:

$$e = \bigsqcup \{c \mid c \in \mathcal{K}(L)\ \&\ c \sqsubseteq e\}$$

A well-known fact is the following: *every algebraic lattice is isomorphic to the family of ideals of a join-semilattice with bottom element.* Let us spell out this statement: a join semilattice with bottom is a partial order such that any finite subset has a join. An ideal is a directed *cone*, that is a directed subset X such that $x \in X\ \&\ y \sqsubseteq x \Rightarrow y \in X$. By a *family* we mean any set of sets, ordered by inclusion.

We can give a more concrete formulation of the last statement, using Scott's notion of information system. Since we are interested in lattices, the information systems we use do not involve the consistency predicate. Moreover, we shall disallows the empty set to enter into consideration. Then our definition is as follows:

DEFINITION (INFORMATION SYSTEMS) 3.9. *An information system is a structure $S = (A, \Delta, \vdash)$ where A is the set of atoms, $\Delta \in A$ and $\vdash$ is a relation over finite non-empty subsets of A satisfying:*

(i) $u \vdash \{\Delta\}$

(ii) $u \vdash u$

(iii) $u \vdash v\ \&\ v \vdash w \Rightarrow u \vdash w$

(iv) $u \subseteq v \Rightarrow v \vdash u$

(v) $u \vdash v\ \&\ u \vdash w \Rightarrow u \vdash v \cup w$

Let us introduce some notations: $\mathit{Fin}(A)$ is the set of *finite non-empty* subsets of A, ranged over by u, v, w... Sometimes we shall denote $u \vdash v$ by $a_1, \ldots, a_n \vdash b_1, \ldots, b_k$ when $u = \{a_1, \ldots, a_n\}$ and $v = \{b_1, \ldots, b_k\}$. For instance $a_1, \ldots, a_n \vdash \Delta$ in any information system.

A basic example is the following: let $(K, \sqsubseteq)$ be a join semilattice with bottom $\perp$. Then it is easy to see that the structure $(K, \perp, \vdash)$ where $\vdash$ is given by

$$a_1, \ldots, a_n \vdash b_1, \ldots, b_k \Leftrightarrow_{\text{def}} (b_1 \sqcup \cdots \sqcup b_k) \sqsubseteq (a_1 \sqcup \cdots \sqcup a_n)$$

is an information structure.

According to Scott, we may understand the atoms $a \in A$ of an information structure as the "propositions" that can be made about the "elements" described by the information structure. The "proposition" Δ means true, and the *deduction* (or entailment) relation $u \vdash v$ formalizes the idea that if the "propositions" of u are true of some element, then the "propositions" of v are also true of the same element. In fact, an element is just the set of propositions true of it. Then the domain of S is the *family* $(\mathcal{D}(S), \subseteq)$ of *non-empty, deductively closed* subsets of A, that is:

$$X \in \mathcal{D}(S) \Leftrightarrow_{\text{def}} \begin{cases} X \subseteq A\ \&\ \Delta \in X \\ u \subseteq X\ \&\ u \vdash v \Rightarrow v \subseteq X \end{cases} \quad \text{and}$$

Remark: for any deductively closed subset X of A we have $X \neq \emptyset \Leftrightarrow \triangle \in X$, since $u \vdash \triangle$ for any $u \in \mathit{Fin}(A)$. For instance, given a non-empty subset X of A, the set

$$\overline{X} =_{\text{def}} \bigcup\{v \mid \exists u \subseteq X.\ u \vdash v\}$$

is an element of $\mathcal{D}(S)$, the least one containing X. Note that this element is also

$$\overline{X} = \bigcap\{U \mid U \in \mathcal{D}(S)\ \&\ X \subseteq U\}$$

In this last formula, the subset X of A can be empty; we shall call this set $\overline{X}$ the *closure* of X. It can be noted that, for $u, v \in \mathit{Fin}(A)$

$$v \subseteq \bar{u} \ \Leftrightarrow\ u \vdash v$$

Returning to our previous basic example, we can see that for a join semilattice with bottom $(K, \sqsubseteq)$, the elements of the domain of the information structure $(K, \bot, \vdash)$ – as defined above – are the *ideals* of the poset $(K, \sqsubseteq)$. More generally, information structures provide us with a concrete presentation of algebraic lattices:

PROPOSITION. *A poset $(L, \sqsubseteq)$ is an algebraic lattice if and only if it is isomorphic to the domain of an information structure. More precisely*

(i) *for any information system the poset $(\mathcal{D}(S), \subseteq)$ is an algebraic lattice, whose compact elements are the sets $\bar{u}$ for $u \in \mathit{Fin}(A)$*

(ii) *any algebraic lattice $(L, \sqsubseteq)$ is isomorphic to the family of ideals of the join semilattice with bottom of its compact elements $(\mathcal{K}(L), \sqsubseteq)$.*

In particular, it is easy to see that for all $X \in \mathcal{D}(S)$

$$X = \bigcup_{u \subseteq X} \bar{u}$$

In $\mathcal{D}(S)$, we can describe the finite joins and meets as follows:

$$X \sqcap Y = X \cap Y \qquad \text{and} \qquad X \sqcup Y = \overline{X \cup Y}$$

We mentioned that the continuous functions between complete lattices form a complete lattice. This lattice is algebraic if the domain and range are algebraic. Then we can give an explicit description of the continuous functions, at the level of information systems: a continuous function is represented by an *approximable mapping* between information systems. The definition, adapted from Scott [43] (see also Coppo & al. 83[15]), is as follows:

DEFINITION (APPROXIMABLE MAPPINGS). *Let $S_i = (A_i, \triangle_i, \vdash_i)$ for $i = 0, 1$ be two information systems. An approximable mapping from S_0 to S_1 is a relation f from $\mathit{Fin}(A_0)$ to $\mathit{Fin}(A_1)$ such that*

(i) $\{\triangle_0\}\, f\, \{\triangle_1\}$

(ii) $u' \vdash_0 u\ \&\ u\, f\, v\ \&\ v \vdash_1 v' \ \Rightarrow\ u'\, f\, v'$

(iii) $u\, f\, v\ \&\ u\, f\, w \ \Rightarrow\ u\, f\, (v \cup w)$

An approximable mapping f from S_0 to S_1 represents a function $\mathsf{fun}(f)\colon \mathcal{D}(S_0) \to \mathcal{D}(S_1)$ given by:

$$\mathsf{fun}(f)(X) =_{\text{def}} \bigcup\{v \mid \exists u \subseteq X.\ u\, f\, v\}$$

This is more or less Plotkin's set-theoretical definition of application (see Barendregt [6]).

For instance it is easy to see that, given $S = (A, \Delta, \vdash)$, the relation $\vdash$ is an approximable mapping over S, and that $\mathsf{fun}(\vdash)$ is the identity function. Remark that for any approximable mapping f we have

$$\mathsf{fun}(f)(\bar{u}) = \bigcup\{v \mid u\ f\ v\}$$

The approximable mappings concretely represent continuous functions:

PROPOSITION. *If f is an approximable mapping from S_0 to S_1 then $\mathsf{fun}(f)$ is a continuous function from $\mathcal{D}(S_0)$ to $\mathcal{D}(S_1)$. Conversely, if g is a continuous function from $\mathcal{D}(S_0)$ to $\mathcal{D}(S_1)$ then the relation $\mathsf{map}(g)$ given by*

$$(u, v) \in \mathsf{map}(g) \Leftrightarrow_{\mathsf{def}} v \subseteq g(\bar{u})$$

is an approximable mapping such that $g = \mathsf{fun}(\mathsf{map}(g))$. This establishes an isomorphism between the poset of continuous functions, extensionally ordered, and the family of approximable mappings, ordered by inclusion.

For instance the compact elements of $(\mathcal{D}(S_0) \to \mathcal{D}(S_1))$, that is the *compact step functions* v/u (with $u \in \mathit{Fin}(A_0)$ and $v \in \mathit{Fin}(A_1)$) given by

$$(v/u)(X) = \begin{cases} \bar{v} & \text{if } \bar{u} \subseteq X \\ \bot & \text{otherwise} \end{cases}$$

are represented by the approximable mappings given as follows:

$$(u', v') \in \mathsf{map}(u/v) \Leftrightarrow v' = \{\Delta\} \quad \text{or} \quad u' \vdash_0 u \ \&\ v \vdash_1 v'$$

We can also describe the general step functions Y/u (with $u \in \mathit{Fin}(A_0)$ and $Y \subseteq A_1$, and, as above, $(Y/u)(X) = \overline{Y}$ if $\bar{u} \subseteq X$):

$$\mathsf{map}(Y/u) = \{\Delta\} \cup \left(\bigcup\{(u', v') \mid u' \vdash_0 u\ \&\ \exists v \subseteq Y\ v \vdash_1 v'\}\right)$$

3.2.2 Construction of the Canonical Interpretation.

In this section we use the information systems to solve the domain equation $D = (D \to D)_\bot$. We have to give two constructions on information systems, respectively representing the lifting construct and the exponentiation – or more accurately the family of approximable mappings.

It is very easy to define the lifting of information structures: let $S = (A, \Delta, \vdash)$ be an information structure, and let us define S' as $(A', \Delta', \vdash')$ where $A' = A \cup \{\Delta'\}$ (with $\Delta' \notin A$), and $\vdash'$ is the least relation containing $\vdash$ and satisfying (i)-(v) of the definition 3.8. Then it is not difficult to see that $\mathcal{D}(S')$ is isomorphic to $(\mathcal{D}(S))_\bot$, and more precisely:

$$X \in \mathcal{D}(S') \Leftrightarrow X = \{\Delta'\} \quad \text{or} \quad \exists Y \in \mathcal{D}(S).\ X = \{\Delta'\} \cup Y$$

We saw that the lifting construct comes with two associated functions

$$\mathsf{up}: D \to D_\bot \quad \text{and} \quad \mathsf{down}: D_\bot \to D$$

Here these functions are given by:

$$\mathsf{down}(X) = \begin{cases} Y & \text{if } X = \{\Delta'\} \cup Y \text{ where } Y \in \mathcal{D}(S) \\ \overline{\{\Delta\}} & \text{otherwise} \end{cases}$$

$$\mathsf{up}(Y) = Y \cup \{\Delta'\}$$

The second construct is the "exponentiation": let $S_i = (A_i, \Delta_i, \vdash_i)$ for $i = 0, 1$ be two information systems. We define

$$(S_0 \to S_1) = (A, \Delta, \vdash)$$

as follows:

(1) A is the set of pairs of finite non-empty subsets of A_0 and A_1, that is $A = \mathbf{Fin}(A_0) \times \mathbf{Fin}(A_1)$

(2) $\Delta = (\{\Delta_0\}, \{\Delta_1\})$

(3) $\vdash$ is the least relation satisfying (i)-(v) of the definition 3.8, and

(3.1) $u' \vdash_0 u \;\&\; v \vdash_1 v' \;\Rightarrow\; (u, v) \vdash (u', v')$

(3.2) $(u, v), (u, w) \vdash (u, v \cup w)$

Note that due to the clause (i) of the definition we have

$$(u_1, v_1), \ldots, (u_n, v_n) \vdash (\{\Delta_0\}, \{\Delta_1\})$$

Then we can show that this is an appropriate representation of the domain of approximable mappings:

LEMMA. *$f \in \mathcal{D}(S_0 \to S_1)$ if and only if f is an approximable mapping from S_0 to S_1.*

A consequence of this result is that the poset $(\mathcal{D}(S_0) \to \mathcal{D}(S_1), \sqsubseteq)$ of continuous functions is isomorphic to $(\mathcal{D}(S_0 \to S_1), \subseteq)$. For instance it can be checked that a step function v/u (identified with its representative approximable mapping) is the element $\overline{w}$ of $\mathcal{D}(S_0 \to S_1)$, where $w = \{(u, v)\}$. Similarly, the identity over $\mathcal{D}(S)$ is $\overline{U}$ with $U = \{(u, u) \mid u \in \mathbf{Fin}(A)\}$.

We are now ready to give the information system describing the canonical domain: intuitively this information system is the least one which is closed for both the lifting and the exponentiation constructs (this is very similar to the Plotkin-Scott-Engeler construction, see Barendregt [6], definition 5.4.5).

DEFINITION (the CANONICAL INFORMATION SYSTEM) 3.10. *The canonical information system is $S_\star = (A_\star, \Delta, \vdash)$ where $A_\star$ is the least set such that*

(i) $\Delta \in A_\star$

(ii) $u, v \in \mathbf{Fin}(A_\star) \;\Rightarrow\; (u, v) \in A_\star$

and $\vdash$ is the least relation on $\mathbf{Fin}(A_\star)$ satisfying the clauses (i)-(v) of the definition 3.8, that is:

$$\begin{aligned}
&\text{D1}: \quad u \vdash \{\Delta\} \\
&\text{D2}: \quad u \vdash u \\
&\text{D3}: \quad u \vdash v \;\&\; v \vdash w \;\Rightarrow\; u \vdash w \\
&\text{D4}: \quad u \subseteq v \;\Rightarrow\; v \vdash u \\
&\text{D5}: \quad u \vdash v \;\&\; u \vdash w \;\Rightarrow\; u \vdash v \cup w
\end{aligned}$$

and also the clauses defining the exponentiation of information systems, that is:

$$\begin{aligned}
&\text{D6}: \quad u \in \mathbf{Fin}(A_\star - \{\Delta\}) \;\Rightarrow\; u \vdash (\{\Delta\}, \{\Delta\}) \\
&\text{D7}: \quad u' \vdash u \;\&\; v \vdash v' \;\Rightarrow\; (u, v) \vdash (u', v') \\
&\text{D8}: \quad (u, v), (u, w) \vdash (u, v \cup w)
\end{aligned}$$

Clearly if we let $D_\star = \mathcal{D}(S_\star)$ then this domain satisfies the equation

$$D_\star = (D_\star \to D_\star)_\perp$$

In $D_\star$ the least element is $\perp = \{\triangle\}$. There is also in $D_\star$ a least element strictly greater than $\perp$, which corresponds to the function whose constant value is $\perp$, namely:

$$\delta =_{\text{def}} \mathsf{up}(\mathsf{down}(\perp)) = \perp \cup \{(\perp, \perp)\}$$

This element, which satisfies:

$$\forall f \in D_\star \; f \neq \perp \;\Leftrightarrow\; \delta \subseteq f$$

will be used in interpreting ∇. We saw that to interpret an application MN we use $\mathsf{down}([\![M]\!]\rho)$, or more precisely the function represented by this element of the domain. Then we will introduce an abbreviation for the application between elements of $D_\star$: for f and e in $D_\star$, we define

$$f \cdot e =_{\text{def}} \mathsf{fun}(\mathsf{down}(f))(e)$$

It is easy to see that the subset $f \cdot e$ of $A_\star$ is the least one such that

(i) $\triangle \in f \cdot e$ and

(ii) $u \subseteq e \;\&\; (u, v) \in f \;\Rightarrow\; v \subseteq f \cdot e$.

Now we proceed to give a concrete interpretation of the λ_J-calculus in the canonical domain. That is, we describe, for each term M and environment $\rho \in \mathit{Env}(D_\star)$, the element $[\![M]\!]\rho$ of $D_\star$ as a subset of $A_\star$ – or more accurately as an union of elements of $\mathit{Fin}(A_\star)$. The first item is $[\![x]\!]\rho = \rho(x)$, which can be formulated pedantically by saying that $[\![x]\!]\rho$ is the least subset of $A_\star$ such that

$$\text{S1}: \quad u \subseteq \rho(x) \;\Rightarrow\; u \subseteq [\![x]\!]\rho$$

where, as before, u ranges over $\mathit{Fin}(A_\star)$.

The concrete formulation of $([\![\lambda x M]\!]\rho)e = [\![M]\!]([x \mapsto e]\rho)$ is a bit more elaborated: the function $g_{x,\rho}$ given by $g_{x,\rho}(e) = [\![M]\!]([x \mapsto e]\rho)$ is, by construction, a continuous function from $D_\star$ to itself, or more precisely the approximable mapping

$$(u, v) \in g_{x,\rho} \;\Leftrightarrow\; v \subseteq [\![M]\!]([x \mapsto \bar{u}]\rho)$$

Then we know how to transform $g_{x,\rho}$ into an element of $D_\star$, namely $\mathsf{up}(g_{x,\rho})$. Therefore we can describe more concretely $[\![\lambda x M]\!]\rho = \mathsf{up}(g_{x,\rho})$ as the least subset of $A_\star$ such that $\triangle \in [\![\lambda x M]\!]\rho$ and

$$\text{S2}: \quad v \subseteq [\![M]\!]([x \mapsto \bar{u}]\rho) \;\Rightarrow\; (u, v) \in [\![\lambda x M]\!]\rho$$

The meaning of application is given by

$$[\![MN]\!]\rho = ([\![M]\!]\rho) \cdot ([\![N]\!]\rho) = \mathsf{fun}(\mathsf{down}([\![M]\!]\rho))([\![N]\!]\rho)$$

therefore we can describe $[\![MN]\!]\rho$ as the least subset of $A_\star$ such that $\triangle \in [\![MN]\!]\rho$ and

$$\text{S3}: \quad u \subseteq [\![N]\!]\rho \;\&\; (u, v) \in [\![M]\!]\rho \;\Rightarrow\; v \subseteq [\![MN]\!]\rho$$

The interpretation of $[\![M \oplus N]\!]\rho$, as the join of $[\![M]\!]\rho$ and $[\![N]\!]\rho$, is quite simple to formulate:

$$\text{S4}: \quad u \in [\![M \oplus N]\!]\rho \;\Leftrightarrow\; u \in \overline{U} \quad \text{where} \quad U = ([\![M]\!]\rho) \cup ([\![N]\!]\rho)$$

Now the interpretation of the observer ∇ is, whatever the environment, the step function $\mathbf{I}_\star/\delta$ where $\mathbf{I}_\star$ (the identity) is the closure of $\{(u,u) \mid u \in \mathit{Fin}(A_\star)\}$. We saw that

$$w \in (\mathbf{I}_\star/\delta) \;\Leftrightarrow\; \begin{cases} \exists u\ \exists v_1,\ldots,v_n.\ u \vdash \delta\ \&\ (v_1,v_1),\ldots,(v_n,v_n) \vdash v\ \&\ w = (u,v) & \text{or} \\ w = \triangle \end{cases}$$

Let us simplify this expression, using the properties D1-D8 to present $[\![\nabla]\!]\rho$ as the closure $\overline{V}$ of V, for some reasonably simple V. First, we remark that $u \vdash \delta \;\Leftrightarrow\; u \vdash (\{\triangle\},\{\triangle\})$ (in one direction one uses D4 and D3, and for the converse one uses D1 and D5). Then, due to D6 and D7, we may assume that $(u,v) \in V \;\Rightarrow\; u = \{(\{\triangle\},\{\triangle\})\}$. Similarly, due to D8 (possibly several times) and D7, we see that we may assume: $(u,v) \in V \;\Rightarrow\; v = (w,w)$ for some $w \in \mathit{Fin}(A_\star)$. Finally we get:

$$\text{S5}: \quad [\![\nabla]\!]\rho = \overline{V} \quad \text{where} \quad V = \{((\bot,\bot),(u,u)) \mid u \in \mathit{Fin}(A_\star)\}$$

Note that in all the previous semantic formulae S1-S5 we have to take only a finite information about the environment into account.

4. The Logical system.

Information systems provide us with a concrete representation for algebraic lattices. In fact one can give an even more concrete presentation by using a *syntax* for information systems. This syntax will be a *logical* one: this formalizes Scott's intuitive explanations about information systems (*cf.* Coppo & al. 83[15], Dezani & Margaria 86[20], and Abramsky 87[1]).

4.1 Logical Interpretation: the Sequent Calculus.

The logical representation of algebraic lattices relies upon a statement *dual* to the one we used in introducing information systems, namely: *every algebraic lattice is isomorphic to the family of filters of a meet-semilattice with top element.* A meet semilattice with top element $\top$ is a poset $(K,\sqsubseteq)$ such that any finite subset has a meet. A filter is a subset X which is filtered (that is: for any finite subset Y of X there exists $x \in X$ such that $x \sqsubseteq \bigsqcap Y$) and satisfies $x \in X\ \&\ x \sqsubseteq y \;\Rightarrow\; y \in X$. Clearly a meet-semilattice with top $(K,\sqsubseteq)$ determines an information structure $(K,\vdash,\top)$ where $\vdash$ is given by:

$$a_1,\ldots,a_n \vdash b_1,\ldots,b_k \Leftrightarrow (a_1 \sqcap \cdots \sqcap a_n) \sqsubseteq (b_1 \sqcap \cdots \sqcap b_k)$$

The domain of this information structure is the family of filters of $(K,\sqsubseteq)$.

The formulae of the logical language associated with a given information system $S = (A,\triangle,\vdash)$ describe the finite non-empty subsets of A, that is elements of $\mathit{Fin}(A)$. We assume given an atomic proposition for each singleton $\{a\}$ where $a \in A$. In particular we shall always use ω to denote $\{\triangle\}$. The union of two elements u and v of $\mathit{Fin}(A)$ respectively denoted by ϕ and ψ will be denoted by $\phi \wedge \psi$, the conjunction of ϕ and ψ. Then the logical language associated with S is the set of formulae given by the grammar:

$$\phi ::= p \mid \omega \mid (\phi \wedge \phi)$$

where p is any atomic proposition.

The next step is to represent the deduction relation $\vdash$. This is turned into an *entailment* relation between formulae, $\phi \leq \psi$ corresponding to $u \vdash v$. Then for instance a deduction

$$a_1, \ldots, a_n \vdash b_1, \ldots, b_k$$

will be denoted by

$$p_1 \wedge \cdots \wedge p_n \leq q_1 \wedge \cdots \wedge q_k$$

Translating the clauses of the definition of information systems (*cf.* also D1-D5 in definition 3.9), we see that the entailment relation has to satisfy the following axioms corresponding to D1-D3:

$$\text{E1}: \ \phi \leq \omega \qquad \text{E2}: \ \phi \leq \phi \qquad \text{E3}: \ \frac{\phi \leq \tau \ , \ \tau \leq \psi}{\phi \leq \psi}$$

Then $\leq$ is a preorder on formulae, and ω is the greatest, or "most general" formula. In categorical terms, we would say that these axioms assert the existence of a terminal object, identities, and composition. To formulate D4 we remark that this requirement is equivalent to $u \cup v \vdash u$. Since the conjunction is not a priori associative and commutative, we shall give two axioms for this clause. Then, together with the axiomatization of D5, we get:

$$\text{E4.1}: \ (\phi \wedge \psi) \leq \phi \qquad \text{E4.2}: \ (\phi \wedge \psi) \leq \psi \qquad \text{E5}: \ \frac{\phi \leq \psi_0 \ , \ \phi \leq \psi_1}{\phi \leq (\psi_0 \wedge \psi_1)}$$

These are the laws for the existence of a product in a preordered set. If we let

$$\phi \sim \psi \ \Leftrightarrow_{\text{def}} \ \phi \leq \psi \ \& \ \psi \leq \phi$$

it is easy to see that

$$\phi \wedge \omega \sim \phi$$
$$\phi \wedge \psi \sim \psi \wedge \phi$$
$$\phi \wedge (\psi \wedge \zeta) \sim (\phi \wedge \psi) \wedge \zeta$$

This allows us to use the notation $\bigwedge_i \phi_i$ for finite conjunctions, up to $\sim$ (with the convention that the empty conjunction is ω). One can also remark for instance that $\phi \sim (\phi \wedge \phi)$.

REMARK 4.1. $\phi_0 \leq \psi_0 \ \& \ \phi_1 \leq \psi_1 \ \Rightarrow \ \phi_0 \wedge \phi_1 \leq \psi_0 \wedge \phi_1$

Finally we have to reflect in the logic the notion of element of the domain, that is deductively closed non-empty subset of A. As we saw above, the appropriate notion is that of a filter of formulae:

DEFINITION (FILTERS). *A filter is a set F of formulae such that*

(i) $\omega \in F$

(ii) $\phi \in F \ \& \ \psi \in F \ \Rightarrow \ \phi \wedge \psi \in F$

(iii) $\phi \in F \ \& \ \phi \leq \psi \ \Rightarrow \ \psi \in F$.

For instance, for any formula ϕ the set $\phi\uparrow = \{\psi \mid \phi \leq \psi\}$ is a (principal) filter, generated by ϕ. We let the reader convinces him/herself that given an information system one can introduce some further axioms for entailment, so that the domain determined by the information system is isomorphic to the family of filters of this logic. We shall only detail the case of $S_\star$.

To deal with the exponentiation construct, we have to give a syntax for the finite non-empty sets of pairs (u, v). Taking advantage of the recursive nature of $A_\star$, we can describe these pairs (or more precisely the singletons consisting of one of these pairs) as implicative formulae $(\phi \to \psi)$. Then the logical language for $\mathbf{Fin}(A_\star)$ is as follows:

$$\phi ::= \omega \mid (\phi \to \phi) \mid (\phi \wedge \phi)$$

We denote by Φ the set of these formulae. Now the axioms for entailment corresponding to the clauses D6-D8 are:

$$\text{E6}: \quad (\phi \to \omega) \leq (\omega \to \omega) \qquad \text{E7}: \quad \frac{\phi_0 \geq \phi_1 \ , \ \psi_0 \leq \psi_1}{(\phi_0 \to \psi_0) \leq (\phi_1 \to \psi_1)}$$

$$\text{E8}: \quad ((\phi \to \phi_0) \wedge (\phi \to \phi_1)) \leq (\phi \to (\phi_0 \wedge \phi_1))$$

Note that E6 is not the exact image of D6, which would be:

$$(\phi_1 \to \psi_1) \wedge \cdots \wedge (\phi_n \to \psi_n) \leq (\omega \to \omega)$$

Let us see that this holds, for $n = 2$:

$$\begin{aligned}
(\phi_0 \to \psi_0) \wedge (\phi_1 \to \psi_1) &\leq ((\phi_0 \wedge \phi_1) \to \psi_0) \wedge ((\phi_0 \wedge \phi_1) \to \psi_1) \quad \text{by E4.i, E7 and remark 4.1} \\
&\leq (\phi_0 \wedge \phi_1) \to (\psi_0 \wedge \psi_1) \qquad \text{by E8} \\
&\leq (\phi_0 \wedge \phi_1) \to \omega \qquad \text{by E1, E7} \\
&\leq \omega \to \omega \qquad \text{by E6}
\end{aligned}$$

One can also remark that a consequence of E1, E6 and E7 is

$$\forall \phi.\ \phi \to \omega \sim \omega \to \omega$$

This logic $(\Phi, \leq)$ is exactly the one of Dezani & Margaria 86[20] (if we forget about the propositional variables, which are not needed since there is a non-trivial initial solution to the canonical domain equation) and of Abramsky 88[2]. To avoid misunderstanding about this logic, we should point out that, although ω means "true" and $\phi \leq \psi$ means "ψ is a consequence of ϕ" (or ψ is more general than ϕ), some usual logical laws do not hold here. For instance $\phi \to \omega \not\sim \omega$, and in fact "$\phi$ implies true" is strictly more informative than "true"; similarly $\omega \to \phi \not\sim \phi$. Note also that $(\phi_0 \wedge \phi_1) \to \phi \not\sim \phi_0 \to (\phi_1 \to \phi)$.

Let us denote by $\mathcal{F}(\Phi, \leq)$ the set of filters of this logic. It should be clear that this provides us with another representation of the canonical domain (*cf.* Coppo & al. 84[14], Abramsky 88[1,2]):

PROPOSITION. *The canonical domain $\mathbf{D}_\star$ is isomorphic to the family $\mathcal{F}(\Phi, \leq)$ of filters (ordered by inclusion).*

In fact we shall regard this statement as giving the – real! – definition of the canonical domain. Consequently, we have to give the – real! – interpretation of the λ_J-calculus in this logical setting. That is, we define $[\![M]\!]\rho$ as a set of formulae (in fact a filter) for each term M and environment $\rho \in \mathbf{Env}(\mathcal{F}(\Phi, \leq))$.

We shall describe $\phi \in [\![M]\!]\rho$ by means of a "typing system", or more accurately a sequent calculus allowing to prove statements of the form $H \vdash M : \phi$ where H is a "logical environment", called *hypothesis*. In this notion of environment $\mathbf{Env}(\Phi)$, "trivial" means that the value is equivalent (with respect to $\sim$) to ω. Thus an hypothesis is a mapping $H : \mathbf{X} \to \Phi$ such that $H(x) \sim \omega$ except for finitely many variables, so that H may be represented as usual as a sequence $x_1 : \phi_1, \ldots, x_n : \phi_n$ (where the variables x such that $H(x) \not\sim \omega$ are among the x_i's). In particular we write $\vdash M : \phi$ for $H \vdash M : \phi$ where $H(x) \sim \omega$ for all x. Therefore the sequent calculus will allow us to prove facts of the form:

$$x_1 : \phi_1, \ldots, x_n : \phi_n \vdash M : \phi$$

to be read: " from the hypothesis that the value of the variables x_i's satisfy the formulae ϕ_i's we may infer that the expression M satisfy ϕ". In fact we shall also use the comma occurring in the hypothesis as a symbol for the updating operation. More precisely, composing elementary updatings we can define, for all hypotheses H and G, the updating of G by H, denoted (H, G), as follows:

$$(H, G)(x) = \begin{cases} H(x) & \text{if } H(x) \not\sim \omega \\ G(x) & \text{otherwise} \end{cases}$$

Then in the notation $x_1 : \phi_1, \ldots, x_n : \phi_n$ a given variable may occur several times – its value is the leftmost one in the sequence.

We shall say that an environment $\rho \in \mathbf{Env}(\mathcal{F}(\Phi, \leq))$, which assigns a set of formulae to a variable, *satisfies* the hypothesis H if $H(x) \in \rho(x)$ for all $x \in X$. To define the sequent calculus we give the logical formulation of the semantic equations S1-S5, guided by the idea that a sequent of the previous form means:

$$\phi_1 \in \rho(x_1), \ldots, \phi_n \in \rho(x_n) \;\Rightarrow\; \phi \in [\![M]\!]\rho$$

The first rules to state assert that $[\![M]\!]\rho$ is a filter, that is:

$$\text{L1}: \quad H \vdash M : \omega \qquad\qquad \text{L2}: \quad \frac{H \vdash M : \phi \;,\; H \vdash M : \psi}{H \vdash M : (\phi \wedge \psi)}$$

$$\text{L3}: \quad \frac{H \vdash M : \phi}{H \vdash M : \psi}\ \phi \leq \psi$$

Note that the usual rules for elimination of conjunction are derived from L3 (using E4.1, E4.2):

$$\frac{H \vdash M : \phi \wedge \psi}{H \vdash M : \phi} \qquad\qquad \frac{H \vdash M : \phi \wedge \psi}{H \vdash M : \psi}$$

Now we give the rules corresponding to terms formation. First, translating S1 and S2 we get:

$$\text{L4}: \quad x : \phi \,,\, H \vdash x : \phi \qquad\qquad \text{L5}: \quad \frac{x : \psi \,,\, H \vdash M : \phi}{H \vdash \lambda x M : (\psi \to \phi)}$$

Note that the exact translation of S2 would be

$$\frac{x:\zeta\ ,\ H\vdash M:\phi}{H\vdash \lambda x M:(\psi\to\phi)}\ \psi\leq\zeta$$

since $w\subseteq\tilde{u}\ \Leftrightarrow\ u\vdash w$. But this is actually derivable from L5, using E7 and L3.

Since a subset of $A_\star$ is represented by a single (conjunctive) formula, it should be clear that we can state the rule for application MN, corresponding to S3, as follows:

$$\text{L6}:\ \frac{H\vdash M:(\phi\to\psi)\ ,\ H\vdash N:\phi}{H\vdash (M\,N):\psi}$$

The inference rules L4-L6 are the usual rules in Curry's typing system.

In $\mathcal{F}(\Phi,\leq)$ the join of two filters is the filter generated by their union (recall that $X\sqcup Y=\overline{X\cup Y}$ in $D_\star$), and $\{\phi\mid H\vdash M:\phi\}$ is always a filter by virtue of L1-L3, therefore we can express S4 by:

$$\text{L7}:\ \frac{H\vdash M:\phi}{H\vdash (M\oplus N):\phi}\qquad\qquad \text{L8}:\ \frac{H\vdash N:\psi}{H\vdash (M\oplus N):\psi}$$

Note that by combining the rules L7, L8 and L2 we can derive

$$\frac{H\vdash M:\phi\ ,\ H\vdash N:\psi}{H\vdash (M\oplus N):\phi\wedge\psi}$$

We presented the interpretation of the observer ∇ (equation S5 above) as the closure of a subset of $A_\star$. Then its logical interpretation is the filter generated (by means of L1-L3) by the corresponding set of formulae, namely:

$$\text{L9}:\ \ H\vdash\nabla:(\omega\to\omega)\to(\phi\to\phi)$$

Some comments about this system are in order: we do not regard the sequent calculus as a *typing* system, because the notion of type usually refers to an idea of *constraint*. Types were introduced to avoid paradoxical situations, and more precisely to rule out self-application. Indeed in any "Curry-Howard" typing system a typable term is strongly convergent (as well as its subterms). Although the relation between the sequent calculus and the computational notion of convergence turns out to be, as usual (see for instance Leivant [27]), a central question, it should be clear that the purpose of "typing" systems à *la* Coppo is not to restrict programs formation, but rather to tell something logical about meaningful programs. Then we could say, following Leivant [27], that Coppo's systems implement an idea of "types as formulae" (but not "programs as proofs"), where $\phi\to\psi$ is an assertion about the functional character of a program.

In what follows, when we write $H\vdash M:\phi$, in most cases we intend that this sequent is provable by means of the previous rules. We let the reader convinces him/herself that the following holds:

PROPOSITION. *For all terms M and environment $\rho\in\mathbf{Env}(\mathcal{F}(\Phi,\leq))$:*

$$(\phi_1\in\rho(x_1),\ldots,\phi_n\in\rho(x_n)\ \Rightarrow\ \phi\in[\![M]\!]\rho)\ \Leftrightarrow\ x_1:\phi_1,\ldots,x_n:\phi_n\vdash M:\phi$$

A consequence of this result is that we gain a new equivalent formulation for the semantical preorder $M \sqsubseteq_\star N$, namely the preorder relative to the sequent calculus, given by:

$$\boxed{M \sqsubseteq_{\mathcal{S}} N \Leftrightarrow_{\text{def}} \forall H\, \forall \phi.\ H \vdash M : \phi \Rightarrow H \vdash N : \phi}$$

Since the rules of the sequent calculus are either "logical", that is independent of the term, or "structural", that is composing assertions on the subterms, it is not surprising that this preorder is a precongruence:

LEMMA. $M \sqsubseteq_{\mathcal{S}} N \Rightarrow \forall C.\ C[M] \sqsubseteq_{\mathcal{S}} C[N]$

PROOF: by a tedious induction on the inference of $H \vdash C[M] : \phi$, omitted □

We shall see later that $\sqsubseteq_{\mathcal{S}}$ is a semantics (*cf.* definition 3.1), that is: this preorder is coherent, contains the β-law $(\lambda x M)N = M[N/x]$, and is computationally adequate.

4.2 Realizability and Soundness.

We also claimed that the canonical semantics – that is now the logical interpretation – is fully abstract:

$$M \sqsubseteq_{\mathcal{O}} N \Rightarrow M \sqsubseteq_{\mathcal{S}} N$$

A first step towards this result consists in giving a "realizability" interpretation of the formulae of Φ. That is, each formula denotes a set of closed terms realizing the given formula. This interpretation is due, for the typed λ-calculus, to Tait. It was originally called "convertibility", and later on received the names of "reducibility" or "computability" (a generalization of this semantics of formulae is credited to Reynolds and Scott in [23,7]). We denote "M realizes ϕ" by $\models M : \phi$, defined as follows (in the definition all the terms are closed):

$$\begin{aligned}
\models M : \omega &\Leftrightarrow_{\text{def}} \text{true} \\
\models M : (\phi \wedge \psi) &\Leftrightarrow_{\text{def}} \models M : \phi \ \&\ \models M : \psi \\
\models M : (\phi \to \psi) &\Leftrightarrow_{\text{def}} M\Downarrow \ \&\ \forall R.\ \models R : \phi \Rightarrow \models MR : \psi
\end{aligned}$$

As one can see, this interpretation, adapted from the one given by Abramsky in [2], is essentially the "F-semantics" of Hindley ([23], see also Dezani & Margaria [20]). The realizability interpretation entirely relies upon the convergence property $M\Downarrow$. In fact it entails a characterization of this property: a closed term is convergent if and only if it realizes the formula $\omega \to \omega$, that is

REMARK 4.2. $M\Downarrow \Leftrightarrow \models M : (\omega \to \omega)$

We can introduce a *logical preorder* on closed terms relative to this interpretation of the formulae: a term is logically less than another one whenever it realizes less formulae, that is

$$M \sqsubseteq^{\circ}_{\mathcal{L}} N \Leftrightarrow_{\text{def}} \forall \phi.\ \models M : \phi \Rightarrow \models N : \phi$$

This preorder is extended to arbitrary terms by instanciation:

$$M \sqsubseteq_{\mathcal{L}} N \Leftrightarrow_{\text{def}} \forall \sigma \text{ closed } M\sigma \sqsubseteq^{\circ}_{\mathcal{L}} N\sigma$$

Now we want to show that this preorder is weaker than the testing preorder. By virtue of the results of section 3.1, it is enough to show:

LEMMA. *If $\mathcal{R}$ is an applicative simulation and $M\,\mathcal{R}\,N$ then for any formula $\phi \in \Phi$ we have* $\models M : \phi \Rightarrow \models N : \phi$.

whose proof (by induction on the formula ϕ) is trivial. As a consequence, we have now:

PROPOSITION 4.3.

$$M \sqsubseteq_{\mathcal{O}} N \Rightarrow M \sqsubseteq_{\mathcal{A}} N \Leftrightarrow M \sqsubseteq_{\mathcal{T}} N \Leftrightarrow M \sqsubseteq_{\mathcal{E}} N \Rightarrow M \sqsubseteq_{\mathcal{L}} N$$

Therefore to prove that the canonical interpretation is fully abstract, it will be enough to prove that $\sqsubseteq_{\mathcal{S}}$ is a semantics such that

$$M \sqsubseteq_{\mathcal{L}} N \Rightarrow M \sqsubseteq_{\mathcal{S}} N$$

Indeed the comparison of $\models M : \phi$ and $\vdash M : \phi$ is the central topic of this work. As usual, this comparison takes the form of a *soundness* and *completeness* result. We shall prove first the soundness of the sequent calculus with respect to an extension of the realizability interpretation. We define $H \models M : \phi$, to be read as "M realizes ϕ under the hypothesis H", as follows: first we say that σ realizes H, in notation $\models \sigma : H$, if σ is closed and $\models \sigma(x) : H(x)$ for any variable x. Then

$$H \models M : \phi \Leftrightarrow_{\text{def}} \forall \sigma \text{ closed. } \models \sigma : H \Rightarrow \models M\sigma : \phi$$

REMARK 4.4. $M \sqsubseteq_{\mathcal{L}} N \Rightarrow \forall H.\ H \models M : \phi \Rightarrow H \models N : \phi$

The soundness property of the sequent calculus is:

$$H \vdash M : \phi \Rightarrow H \models M : \phi$$

The argument is the standard one: this implication is proved by induction on the inference of the sequent $H \vdash M : \phi$. To this end we show the validity of each rule of the logical system. The validity of L1 and L2 is trivial. The validity of L3 results from the following:

LEMMA 4.5. *If* $\models M : \phi$ *and* $\phi \leq \psi$ *then* $\models M : \psi$.

PROOF: by induction on the definition of $\phi \leq \psi$, straightforward ❑

The validity of L4 results from the definition of $H \models M : \phi$.

LEMMA 4.6. *If* $x : \psi$, $H \models M : \phi$ *then* $H \models \lambda x M : \psi \to \phi$.

PROOF: let σ be a substitution such that $\models \sigma : H$, and assume that R is a closed term such that $\models R : \psi$. Then the substitution $\sigma' = [x \mapsto R]\sigma$ is such that $\models \sigma' : (x : \psi, H)$, hence $\models M\sigma' : \phi$. By definition of the substitution

$$(\lambda x M)\sigma = \lambda y(M([x \mapsto y]\sigma))$$

for some new variable y. Let $N = M([x \mapsto y]\sigma)$; then by the proposition 2.1 and lemma 2.2

$$M\sigma' \equiv N[R/y]$$

therefore $\models N[R/y] : \phi$. Moreover, by the β-rule:

$$(\lambda y N)R \twoheadrightarrow N[R/y]$$

therefore by the lemma 3.5 and proposition 4.3 above $\models (\lambda y N)R : \phi$, that is $\models ((\lambda x M)\sigma)R : \phi$. Since $(\lambda x M)\sigma \in \boldsymbol{N}$ we have $(\lambda x M)\sigma\Downarrow$, therefore $H \models \lambda x M : (\psi \to \phi)$ ❑

By definition of the interpretation of $(\phi \to \psi)$, the rule L6 is valid. The validity of L7 and L8 should be clear: since $(M \oplus N) \Rightarrow M$ and $(M \oplus N) \Rightarrow N$ we have (by the lemma 3.5 and proposition 4.3) $M \sqsubseteq_{\mathcal{L}} (M \oplus N)$ and $N \sqsubseteq_{\mathcal{L}} (M \oplus N)$. Therefore by the remark 4.4

$$H \models M : \phi \;\Rightarrow\; H \models (M \oplus N) : \phi \qquad \text{and} \qquad H \models N : \psi \;\Rightarrow\; H \models (M \oplus N) : \psi$$

The last step towards the soundness of the logical system consists in showing the validity of L9, that is:

LEMMA 4.7. $H \models \nabla : (\omega \to \omega) \to (\phi \to \phi)$

PROOF: assume that M is a closed term such that $\models M : \omega \to \omega$. Then by the remark 4.2 $M\Downarrow$, hence by the corollary 2.6, $(\nabla M) \Rightarrow \mathbf{I}$. Moreover from the lemma 3.5, $\mathbf{I} \sqsubseteq_{\mathcal{E}} (\nabla M)$, hence by proposition 4.3, $\models \mathbf{I} : \psi \;\Rightarrow\; \models (\nabla M) : \psi$. It is easily established that $\models \mathbf{I} : \phi \to \phi$ for any ϕ, therefore

$$\models M : \omega \to \omega \;\Rightarrow\; \models (\nabla M) : \phi \to \phi$$

that is $\models \nabla : (\omega \to \omega) \to (\phi \to \phi)$ since $\nabla\Downarrow$. Finally since ∇ is closed, we have

$$H \models \nabla : (\omega \to \omega) \to (\phi \to \phi)$$

for any H □

Summarizing, we have established:

PROPOSITION (SOUNDNESS). *If* $H \vdash M : \phi$ *then* $H \models M : \phi$.

4.3 Main Results (Full Abstraction).

As announced, we shall prove a completeness theorem, that is:

$$\boxed{H \models M : \phi \;\Rightarrow\; H \vdash M : \phi}$$

An obvious consequence of soundness and completeness (and of the proposition 4.3 and remark 4.4) is a first half of the full abstraction result, namely that the testing preorder is stronger than the semantical preorder:

$$M \sqsubseteq_{\mathcal{O}} N \;\Rightarrow\; M \sqsubseteq_{\mathcal{S}} N$$

To prove the converse, we have to prove that $\sqsubseteq_{\mathcal{S}}$ is a semantics, that is satisfies the β-law (a point that will be seen later) and computational adequacy. As a matter of fact this last property, that is

$$M, N \text{ closed } \&\; M \sqsubseteq_{\mathcal{S}} N \;\Rightarrow\; (M\Downarrow \;\Rightarrow\; N\Downarrow)$$

is an easy consequence of the soundness and completeness theorems: if $M\Downarrow$ then $\models M : \omega \to \omega$ (remark 4.2), hence by completeness $\vdash M : \omega \to \omega$. Then using the hypothesis $M \sqsubseteq_{\mathcal{S}} N$ we have $\vdash N : \omega \to \omega$, therefore by soundness $\models N : \omega \to \omega$, that is $N\Downarrow$.

Now we indicate what are the main points in proving the completeness theorem. First we will show some structural properties, namely weakening, cut and paste. Our weakening result is somewhat stronger than the usual one. It relies upon an extension of the entailment relation to hypotheses, relative to a given set of variables V. Let us define:

$$G \leq_V H \;\Leftrightarrow_{\text{def}}\; \forall x \in V.\; G(x) \leq H(x)$$

We shall omit the subscript when it is $\mathcal{X}$. For instance $(G,H) \leq G$ for any G and H. Then the *weakening* lemma – which would perhaps better be named "strengthening" ([5]) – asserts that if under the hypothesis H we can prove that M satisfies ϕ, then by strengthening the hypothesis, we are able to prove the same fact. Formally, if $\mathsf{fv}(M) \subseteq V$ then

$$\boxed{H \vdash M : \phi \;\&\; G \leq_V H \;\Rightarrow\; G \vdash M : \phi}$$

Then we have the usual *cut* property, that is the validity of the rule:

$$\frac{H \vdash N : \psi \quad , \quad x : \psi, H \vdash M : \phi}{H \vdash M[N/x] : \phi}$$

We shall prove more generally:

$$\boxed{\left.\begin{array}{r} H \vdash N_1 : \phi_1, \ldots, H \vdash N_k : \phi_k \\ x_1 : \phi_1, \ldots, x_k : \phi_k \vdash M : \phi \end{array}\right\} \;\Rightarrow\; H \vdash M[N_1/x_1 \cdots N_k/x_k] : \phi}$$

In Coppo's sequent calculi a kind of converse property, which we call *paste*, also holds. This property says that a proof of $H \vdash M\sigma : \phi$ is always composed of proofs of intermediary assertions for the $\sigma(x)$'s and a proof that M satisfies ϕ under the corresponding hypothesis. Formally, if $\mathsf{fv}(M) = \{x_1, \ldots, x_k\}$ then

$$\boxed{H \vdash M[N_1/x_1 \cdots N_k/x_k] : \phi \;\Rightarrow\; \exists \phi_1, \ldots, \phi_k \left\{\begin{array}{l} H \vdash N_1 : \phi_1, \ldots, H \vdash N_k : \phi_k \\ x_1 : \phi_1, \ldots, x_k : \phi_k \vdash M : \phi \end{array}\right.}$$

These properties will allow us to show the validity of the β-law:

$$(\lambda x M)N \simeq_{\mathcal{S}} M[N/x]$$

Then we shall prove a result saying that the computational property of convergence (for arbitrary terms) can be ascertained using the sequent calculus. To this end, let us introduce some notations: we write $M\downarrow_H$ for $H \vdash M : \omega \to \omega$ and $M\Downarrow_H$ if $M\sigma\Downarrow$ for any closed substitution σ such that $\models \sigma : H$ (that is $M\Downarrow_H \;\Leftrightarrow\; H \models M : \omega \to \omega$). We will show:

CONVERGENCE LEMMA.

$$\boxed{M\Downarrow_H \;\Rightarrow\; M\downarrow_H}$$

The proof of this lemma uses two auxiliary results, the first one being:

CONVERGENCE LEMMA, CLOSED TERMS. *If M is a closed term then $M\Downarrow \;\Rightarrow \vdash M : \omega \to \omega$*

([5]) the usual names of the structural properties refer to a "bottom-up" search for a proof of a given sequent. If we think of a "top-down" construction of proofs, we should find dual names for these properties, exchanging for instance cut and paste.

The crucial fact allowing us to prove the convergence lemma is the definability of the compact elements of the canonical domain – which are now the principal filters, generated by a formula, that is $\{\psi \mid \phi \leq \psi\}$. Then the definability result can be stated as follows:

CHARACTERIZATION LEMMA. *For any formula ϕ there exists a closed term $\mathbf{M}_\phi$ such that*

$$\vdash \mathbf{M}_\phi : \psi \;\Leftrightarrow\; \phi \leq \psi$$

This is the only result for which we need to have the observer ∇ and the join $\oplus$ in the syntax([6]). To define the characteristic terms $\mathbf{M}_\phi$ we associate with each formula ϕ a "test for ϕ", that is a closed term $\mathbf{T}_\phi$ such that $\models M : \phi \;\Rightarrow\; \mathbf{T}_\phi M \stackrel{*}{\longrightarrow} \mathbf{I}$. The construction of these terms is as follows:

$$\begin{aligned}
\mathbf{M}_\omega &= \Omega \\
\mathbf{T}_\omega &= \mathbf{F} = \lambda y \lambda x x \\
\\
\mathbf{M}_{\phi\wedge\psi} &= (\mathbf{M}_\phi \oplus \mathbf{M}_\psi) \\
\mathbf{T}_{\phi\wedge\psi} &= \lambda x (\mathbf{T}_\phi x)(\mathbf{T}_\psi x) \\
\\
\mathbf{M}_{\phi\to\psi} &= \lambda x (\mathbf{T}_\phi x)\mathbf{M}_\psi \\
\mathbf{T}_{\phi\to\psi} &= \lambda x (\nabla x)(\mathbf{T}_\psi (x\mathbf{M}_\phi))
\end{aligned}$$

We can now prove that the convergence lemma holds:

PROOF of the CONVERGENCE LEMMA: Assume that $M\Downarrow_H$, and let χ be the (closed) substitution given by $\chi(x) = \mathbf{M}_{H(x)}$. Then, since $\models \chi : H$ by the characterization lemma above and soundness, we have $M\chi\Downarrow$, hence, by the convergence lemma for closed terms:

$$\vdash M\chi : \omega \to \omega$$

Let $\mathsf{fv}(M) = \{x_1, \ldots, x_k\}$; then by paste there exist $\phi_1, \ldots, \phi_k$ such that

$$x_1 : \phi_1, \ldots, x_k : \phi_k \vdash M : \omega \to \omega$$

and $\vdash \chi(x_i) : \phi_i$. Then, by the characterization lemma, $H(x_i) \leq \phi_i$ for $x_i \in \mathsf{fv}(M)$, therefore by weakening $H \vdash M : \omega \to \omega$ □

Finally to prove the completeness theorem, we use Hindley's method [23,24], by means of a refinement of the usual deduction theorem, namely:

EXTENSIONALITY (THE DEDUCTION THEOREM).

$$H \vdash M : \phi \to \psi \;\Leftrightarrow\; M\downarrow_H \quad \text{and} \quad x : \phi, H \vdash Mx : \psi \qquad \text{for some } x \notin \mathsf{fv}(M)$$

Dezani and Margaria showed in [20] that such a property was needed for completeness: they added to the "typing" system a rule, called Hindley's rule, whose typical instance is

$$\frac{H \vdash M : \phi \wedge (\omega \to \omega)}{H \vdash \lambda x(Mx) : \phi} \quad x \notin \mathsf{fv}(M)$$

([6]) the reader will note however that we only use the join of finite terms.

They also noted that this rule is not valid if the logical language contains propositional variables.

PROOF of the COMPLETENESS THEOREM: we show $H \models M : \phi \Rightarrow H \vdash M : \phi$ by induction on the formula ϕ. The cases where $\phi = \omega$ and $\phi = \phi_0 \wedge \phi_1$ are obvious (using L1 and L2).

If $H \models M : \phi \to \psi$ then $\models M\sigma : \phi \to \psi$ for any closed substitution σ such that $\models \sigma : H$, hence $M\Downarrow_H$, and $\models (M\sigma)R : \psi$ for any closed term R such that $\models R : \phi$. Since $H \models M : \omega \to \omega$ we have $M\downarrow_H$ by the convergence lemma. Let x be a variable not in $\mathsf{fv}(M)$. Then by proposition 1.1 (i)

$$(M\sigma)R = (Mx)([x \mapsto R]\sigma)$$

hence $x : \phi$, $H \models Mx : \psi$ by definition. By induction hypothesis

$$x : \phi, H \vdash Mx : \psi$$

therefore $H \vdash M : \phi \to \psi$ by the extensionality lemma ❑

REMARK. It should be noted that this result holds for any extension $\Lambda(S)$ of the λ-calculus, where S is a set of closed terms of $\Lambda_J(\nabla)$, satisfying the characterization lemma. Abramsky has shown that this is the case if S contains ∇ and a "minimal" parallel convergence testing combinator, which can be defined by:

$$P = \lambda xy.((\nabla x)\lambda z\Omega \oplus (\nabla y)\lambda z\Omega)$$

It remains to prove the results mentioned above, that is: weakening, cut and paste, validity of β, extensionality (the deduction theorem), convergence lemma for closed terms and the characterization lemma. This will be done in the next section, but we first give some consequences of completeness and full abstraction.

4.4 Some Consequences.

A first obvious consequence of full abstraction is that all the previously encountered preorders coincide. Then we can forget about the subscripts, and simply write $M \sqsubseteq N$ and $M \simeq N$. Here we list, without proving them, some properties of this preorder. It is a precongruence, and in particular we have:

$$(\mu^*) \quad M \sqsubseteq M' \Rightarrow NM \sqsubseteq NM'$$
$$(\nu^*) \quad M \sqsubseteq M' \Rightarrow MN \sqsubseteq M'N$$
$$(\xi^*) \quad M \sqsubseteq M' \Rightarrow \lambda xM \sqsubseteq \lambda xM'$$

We already mentioned the property of weak extensionality, that is:

$$(\eta^-) \quad M \sqsubseteq \lambda x(Mx) \qquad \text{where } x \notin \mathsf{fv}(M)$$

and

$$(\eta^*) \quad M\Downarrow \Rightarrow M \simeq \lambda x(Mx) \qquad \text{where } x \notin \mathsf{fv}(M)$$

The ω-rule of the λ-calculus (see Barendregt [6] definition 4.1.10) is not valid: for instance we have $\Omega R \simeq (\lambda x\Omega)R$ for all closed terms R, but $\Omega \not\simeq \lambda x\Omega$. However we have, for closed terms:

$$(M \prec N \;\&\; \forall R \text{ closed}.\ MR \sqsubseteq NR) \Rightarrow M \sqsubseteq N$$

We could regard the terms $\mathbf{M}_\phi$ such that $\mathbf{M}_\phi \sqsubseteq M$ as the (finite) *approximants* of M. Clearly the following holds, for closed terms:

$$\models M : \phi \Leftrightarrow \mathbf{M}_\phi \sqsubseteq M$$

and

$$M \sqsubseteq N \;\Leftrightarrow\; \forall \phi.\, \mathbf{M}_\phi \sqsubseteq M \;\Rightarrow\; \mathbf{M}_\phi \sqsubseteq N$$

One can also note that the axiomatization of entailment (E1-E8) is *complete* (*cf.* lemma 4.8), that is:

$$\begin{aligned} \phi \leq \psi \;&\Leftrightarrow\; \mathbf{M}_\phi \sqsubseteq \mathbf{M}_\psi \\ &\Leftrightarrow\; \forall M.\; \models M : \phi \;\Rightarrow \models M : \psi \end{aligned}$$

We can easily adapt Abramsky's property of *approximability*, saying that to get a finite information about an application MN we only need a finite information about the operator and a finite information about the operand:

$$\forall \phi \not\sim \omega.\quad \mathbf{M}_\phi \sqsubseteq MN \;\Leftrightarrow\; \exists \psi.\, \mathbf{M}_\psi \sqsubseteq N \;\&\; \mathbf{M}_{\psi \to \phi} \sqsubseteq M$$

Note also the full testing ability is achieved using only *finite* tests, namely $C_\phi = (\mathbf{M}_{\phi \to (\omega \to \omega)}[])$ since we have

$$\models M : \phi \;\Leftrightarrow\; C_\phi[M]\Downarrow$$

Some obvious properties of the join (namely the ones of set-theoretic union) are:

$$\begin{aligned} (M \oplus (N \oplus P)) &\simeq ((M \oplus N) \oplus P) \\ (M \oplus N) &\simeq (N \oplus M) \\ (M \oplus M) &\simeq M \\ (M \oplus \Omega) &\simeq M \end{aligned}$$

Obviously $(M \oplus N)$ is the least upper bound of M and N:

$$\forall P.\; M \sqsubseteq P \;\&\; N \sqsubseteq P \;\Leftrightarrow\; (M \oplus N) \sqsubseteq P$$

All the constructs of the λ_J-calculus distribute over the join – in other words, they are linear:

$$\begin{aligned} (M \oplus N)P &\simeq MP \oplus NP \\ M(N \oplus P) &\simeq MN \oplus MP \\ \lambda x(M \oplus N) &\simeq (\lambda x M \oplus \lambda x N) \end{aligned}$$

For instance

$$\mathbf{K} \oplus \mathbf{F} \simeq \lambda xy.(x \oplus y)$$

We indicated in the introduction how to define a parallel disjunction (parallel or):

$$\mathbf{O} =_{\text{def}} \lambda xy.((x\mathbf{T})y \oplus ((y\mathbf{T})x) = (\mathbf{V}_l \oplus \mathbf{V}_r)$$

One can check that this combinator satisfies the required properties, namely:

$$\begin{aligned} (\mathbf{O}\Omega)\mathbf{T} &\simeq \mathbf{T} \simeq (\mathbf{O}\mathbf{T})\Omega \\ (\mathbf{O}\mathbf{F})\mathbf{F} &\simeq \mathbf{F} \end{aligned}$$

The parallel convergence testing combinator, defined by

$$\mathbf{P} =_{\text{def}} \lambda xy.((\nabla x) \oplus (\nabla y)) \simeq \lambda xy.\nabla(x \oplus y)$$

satisfies its specification (see Abramsky and Ong [2,30,3]), that is:

$$\mathbf{P}MN\Downarrow \Leftrightarrow \mathbf{P}MN \stackrel{*}{\Longrightarrow} \mathbf{I} \Leftrightarrow M\Downarrow \text{ or } N\Downarrow$$

We can also associate with each combinator M (i.e. closed term) a "strict" version of it, given by $M^s = \lambda x(\nabla x)Mx$, which satisfies

$$M^sN \simeq \begin{cases} MN & \text{if } N\Downarrow \\ \Omega & \text{otherwise} \end{cases}$$

5. The Logical System: Completeness.

5.1 Structural Properties: Weakening and Cut.

Usually in a logical calculus there is no particular "annotation" associated with the formulae: a sequent has the form $\Gamma \vdash \phi$ where $\Gamma = \phi_1, \ldots, \phi_n$ is a sequence of formulae. In fact this sequence is almost always a multiset, since one usually assumes the *exchange* rule:

$$\frac{\Gamma, \phi, \psi, \Delta \vdash \zeta}{\Gamma, \psi, \phi, \Delta \vdash \zeta}$$

However, in our system we can only exchange two items $x:\phi$ and $y:\psi$ in the hypothesis if the variables are distinct. For instance $x:\phi, x:\psi \vdash x:\phi$, but the (generally not provable) sequent $x:\psi, x:\phi \vdash x:\phi$ would lead to the wrong conclusion $\vdash \lambda x\lambda x x:\phi \to (\psi \to \phi)$. Similarly, the *contraction* rule

$$\frac{\Gamma, \phi, \phi, \Delta \vdash \zeta}{\Gamma, \phi, \Delta \vdash \zeta}$$

does not really make sense in our system: obviously we may contract $x:\phi, x:\phi$ into $x:\phi$, but it would be absurd to try to contract $x:\phi, y:\phi$ if $x \neq y$. Another rule which is frequently assumed (and, indeed, needed to "type" λ-terms such as **K**) is the *weakening* rule:

$$\frac{\Gamma \vdash \phi}{\Gamma, \Delta \vdash \phi}$$

This rule holds in our sequent calculus, and more generally we have:

LEMMA (WEAKENING) 5.1. *If $H \vdash M:\phi$ and $G \leq_V H$ with $\mathsf{fv}(M) \subseteq V$ then $G \vdash M:\phi$.*

PROOF: by induction on the inference of the sequent $H \vdash M:\phi$, and by case on the last rule used in this inference:

- this is obvious for L1 and L9. For L2 and L3, we simply use the induction hypothesis.
- for L4 we have $M = x$, hence $x \in V$. Then $G \leq_V (x:\phi, H)$ implies $G = (x:\psi, F)$ with $\psi \leq \phi$ (and $F \leq_{V-\{x\}} H$). By L4 and L3 we then have $G \vdash x:\phi$.
- for L5 $M = \lambda x N$ and $\phi = (\xi \to \psi)$ with $x:\xi, H \vdash N:\psi$. Let $W = \{x\} \cup V$; then $\mathsf{fv}(N) \subseteq W$ and $(x:\xi, G) \leq_W (x:\xi, H)$, hence by induction hypothesis $x:\xi, G \vdash N:\psi$ and we use L5 to conclude.
- for L6 we have $M = (PQ)$ with $H \vdash P:(\psi \to \phi)$ and $H \vdash Q:\psi$. Since $\mathsf{fv}(P) \subseteq \mathsf{fv}(M)$ and $\mathsf{fv}(Q) \subseteq \mathsf{fv}(M)$ we have by induction hypothesis $G \vdash P:(\psi \to \phi)$ and $G \vdash Q:\psi$, and we use L6 to infer $G \vdash M:\phi$. The argument is similar for L7 and L8 ❑

Let us see some instances of this lemma: we saw that $(H, F) \leq H$, therefore the usual weakening rule

$$H \vdash M:\phi \Rightarrow H, F \vdash M:\phi$$

is valid. This lemma also allows us to "restrict" the hypothesis to the free variables of the subject term M (assigning Ω to any other variable), or to introduce conjunction into the hypotheses: given two hypotheses F and G we define their conjunction $F \wedge G$ by $(F \wedge G)(x) = F(x) \wedge G(x)$ for all x. The following fact is an easy consequence of E4.1, E4.2 and E5:

REMARK. $H \leq_V F \wedge G \Leftrightarrow H \leq_V F \;\&\; H \leq_V G$

Therefore we have:

COROLLARY 5.2. *If $F \vdash M : \phi$ or $G \vdash M : \phi$ then $F \wedge G \vdash M : \phi$.*

Another structural rule (in the usual logical sense, that is concerning the structure of the hypotheses) which should hold in any sequent calculus is the *cut* rule, whose usual formulation is:

$$\frac{\Gamma \vdash \psi \qquad \psi, \Gamma \vdash \phi}{\Gamma \vdash \phi}$$

Let us introduce a notation (similar to $\models \sigma : H$): given a set V of variable, a substitution σ (not necessarily closed) satisfy the hypothesis H on V under the hypothesis G, in notation $G \vdash_V \sigma : H$ if $G \vdash \sigma(x) : H(x)$ for all $x \in V$. We omit the subscript when it is $\boldsymbol{X}$. In our system, the cut rule will result from:

PROPOSITION 5.3. *If $H \vdash M : \phi \;\&\; G \vdash \sigma : H \;\Rightarrow\; G \vdash M\sigma : \phi$.*

PROOF: by induction on the inference of the sequent $H \vdash M : \phi$, and by case on the last rule used in this inference. The only case deserving some consideration is L5. In this case we have $M = \lambda x N$ and $\phi = (\zeta \to \psi)$ with $x : \zeta, H \vdash N : \psi$. By definition of substitution $M\sigma = \lambda y(N([x \mapsto y]\sigma))$ for some new variable y. Let $\rho = [x \mapsto y]\sigma$. Then by L4 $y : \zeta, G \vdash \rho(x) : \zeta$, and, by weakening, $y : \zeta, G \vdash \rho(z) : H(z)$ for $z \in \mathsf{fv}(M)$ since $y \notin \mathsf{fv}(\rho(z))$. Therefore by induction hypothesis

$$y : \xi, G \vdash N\rho : \phi$$

hence $G \vdash M\sigma : \phi$ by L5 □

COROLLARY (CUT) 5.4. $H \vdash N : \psi \;\;\&\;\; x : \psi, H \vdash M : \phi \;\Rightarrow\; H \vdash M[N/x] : \phi$

PROOF: recall that $M[N/x]$ is $M\sigma$ where $\sigma(x) = N$, $\sigma(y) = y$ for $y \in \mathsf{fv}(M) - \{x\}$ and $\sigma(z) = \Omega$ for any other z. Let F be the restriction of H to M, that is

$$F(y) = \begin{cases} H(y) & \text{if } y \in \mathsf{fv}(M) \\ \omega & \text{otherwise} \end{cases}$$

Then by weakening $x : \psi, F \vdash M : \phi$, and $H \vdash \sigma : (x : \psi, F)$, therefore by the previous proposition $H \vdash M\sigma : \phi$ □

5.2 Extensionality, Paste and Reduction.

In this section we first prove a refined version of the deduction theorem. The usual statement of the deduction theorem, in logical systems without "annotations" is:

$$\Gamma \vdash \phi \to \psi \;\Leftrightarrow\; \phi, \Gamma \vdash \psi$$

It is very easy to prove that this holds in our sequent calculus: in the "$\Leftarrow$" direction, this is just L5. For the converse, if $H \vdash M : \phi \to \psi$ and x is a variable not in $\mathsf{fv}(M)$ then by weakening $x : \phi, H \vdash M : \phi \to \psi$ and $x : \phi, H \vdash x : \phi$ by L4, therefore $x : \phi, H \vdash Mx : \psi$ by L6 (modus ponens). However if we abstract once more we get the term $\lambda x M x$ which is usually more informative than M (for instance if $M = \Omega$). As indicated above, we need a more refined result. Recall that the notation $M{\downarrow_H}$ means $H \vdash M : \omega \to \omega$. Then our sharpened deduction theorem is:

EXTENSIONALITY LEMMA (THE DEDUCTION THEOREM) 5.5. *$H \vdash M : \phi \to \psi$ if and only if $M\downarrow_H$ and $x : \phi, H \vdash Mx : \psi$ for some $x \notin \mathsf{fv}(M)$.*

PROOF: if $H \vdash M : \phi \to \psi$ then we have $H \vdash M : \omega \to \omega$ (that is $M\downarrow_H$) by L3 since by E1, E7 and E6

$$\phi \to \psi \leq \phi \to \omega \leq \omega \to \omega$$

Moreover for $x \notin \mathsf{fv}(M)$ we have by weakening $x : \phi, H \vdash M : \phi \to \psi$ and $x : \phi, H \vdash x : \phi$ by L4. Then by L6 we get $x : \phi, H \vdash Mx : \psi$.

Conversely, assume that $M\downarrow_H$ and $x : \phi, H \vdash Mx : \psi$ for some $x \notin \mathsf{fv}(M)$. We prove that $H \vdash M : \phi \to \psi$ by induction on the inference of $x : \phi, H \vdash Mx : \psi$:

- if this sequent is proved using L1 then $\psi = \omega$, and $H \vdash M : (\phi \to \omega)$ by L3 since $M\downarrow_H$ (that is $H \vdash M : \omega \to \omega$) and $\omega \to \omega \leq \phi \to \omega$ (by E1 and E7)
- by L2, we have $\psi = \psi_0 \wedge \psi_1$, and by induction hypothesis $H \vdash M : \phi \to \psi_i$. Then by L2 $H \vdash M : (\phi \to \psi_0) \wedge (\phi \to \psi_1)$, hence by L3 and E8 we get $H \vdash M : \phi \to (\psi_0 \wedge \psi_1)$.
- by L3, $x : \phi, H \vdash Mx : \psi'$ for some ψ' such that $\psi' \leq \psi$. Then by induction hypothesis $H \vdash M : \phi \to \psi'$, hence $H \vdash M : \phi \to \psi$ by L3 since $\phi \to \psi' \leq \phi \to \psi$ (by E7).
- the only remaining case is L6. Then for some ϕ' we have $x : \phi, H \vdash M : \phi' \to \psi$ and $x : \phi, H \vdash x : \phi'$. By the previous lemma $\phi \leq \phi'$, therefore by L3 $x : \phi, H \vdash M : \phi \to \psi$ since $\phi' \to \psi \leq \phi \to \psi$ (by E7). Since $x \notin \mathsf{fv}(M)$ we have $H \vdash M : \phi \to \psi$ by weakening ❑

An important property of Coppo's typing systems is that if a term M reduces into N and N satisfies ϕ under the hypothesis H, then the same holds for M (*cf.* [12,26]). To establish this property in the case of the β-rule $(\lambda x M)N \Rightarrow M[N/x]$, we need to know how to "type" $M[N/x]$. This is expressed in the paste property, proved below. Let us first show that we cannot prove more about a variable than assumed:

LEMMA 5.6. $H \vdash x : \phi \Leftrightarrow H(x) \leq \phi$

PROOF: by induction on the inference of the sequent $H \vdash x : \phi$ (which can only be proved by means of L1, L2, L3 or L4), trivial (one uses E1, E2, E3 and E5) ❑

PROPOSITION (PASTE) 5.7. *Let $\mathsf{fv}(M) = V$; then*

$$H \vdash M\sigma : \phi \;\Rightarrow\; \exists G.\ H \vdash_V \sigma : G \;\&\; G \vdash M : \phi$$

PROOF: by induction on the proof of the sequent $H \vdash M\sigma : \phi$, and then by case on M. We first treat the case where M is a variable, say x: in this case the proposition is trivial since $M\sigma = \sigma(x)$ and $x : \phi \vdash x : \phi$. Then for the rest of the proof we shall assume that M is not a variable (so we will not consider the case where the sequent $H \vdash M\sigma : \phi$ is proved using L4).

- if the sequent $H \vdash M\sigma : \phi$ is proved using L1, we let $G(x) = \omega$ for all x. The proof is trivial if the sequent is deduced by L3 or L9.
- by L2, we have $\phi = \phi_0 \wedge \phi_1$ with $H \vdash M\sigma : \phi_i$ for $i = 0, 1$. By induction hypothesis there exist G_0 and G_1 such that $H \vdash_V \sigma : G_i$ and $G_i \vdash M : \phi_i$. We let $G = G_0 \wedge G_1$; then $H \vdash_V \sigma : G$ by L2 and $G \vdash M : \phi_i$ by the corollary 5.2, hence $G \vdash M : \phi$ by L2.
- by L5, we have $\phi = (\zeta \to \psi)$ and $M\sigma = \lambda y P$ with $y : \zeta, H \vdash P : \psi$. Since M is not a variable, we have $M = \lambda x N$ with $P = N([x \mapsto y]\sigma)$ where y is a new variable. Let $\rho = [x \mapsto y]\sigma$; then by induction hypothesis there exist G such that

(1) $$y : \zeta, H \vdash_U \rho : G$$

where $U = \mathsf{fv}(N)$, and

$$G \vdash N : \psi \tag{2}$$

Then from (1) we have $y:\zeta$, $H \vdash_{U-\{x\}} \sigma : G$, therefore by weakening $H \vdash_V \sigma : G$ where $V = U - \{x\} = \mathsf{fv}(M)$ since $z \in V \Rightarrow y \notin \mathsf{fv}(\sigma(z))$. If $x \notin U$ then from (2) by weakening $x:\zeta$, $G \vdash N : \psi$, hence $G \vdash M : \phi$ by L5. On the other hand, if $x \in U$ then from (1) we have $y:\zeta$, $H \vdash_U y : G(x)$, hence by the lemma 5.6, $\zeta \leq G(x)$. Then by weakening from (2) we get $x:\zeta$, $G \vdash N : \psi$, therefore $G \vdash M : \phi$ by L5.

• by L6, we have, since M is not a variable, $M = (PQ)$ with $H \vdash P\sigma : (\psi \to \phi)$ and $H \vdash Q\sigma : \psi$ for some ψ. By induction hypothesis, there exist F and G such that $H \vdash_U \sigma : F$ where $U = \mathsf{fv}(P)$ and $F \vdash P : (\psi \to \phi)$, and $H \vdash_W \sigma : G$ where $W = \mathsf{fv}(Q)$ and $G \vdash Q : \psi$. Let E be the hypothesis given by $E(x) = \omega$ if $x \notin U \cup W$ and

$$E(x) = \begin{cases} F(x) & \text{if } x \in U - W \\ F(x) \wedge G(x) & \text{if } x \in U \cap W \\ G(x) & \text{if } x \in W - U \end{cases}$$

Then using possibly L2 we see that $H \vdash_V \sigma : E$ where $V = U \cup W = \mathsf{fv}(M)$, and by weakening $E \vdash P : (\psi \to \phi)$ and $E \vdash Q : \psi$, therefore $E \vdash M : \phi$ by L6.

• by L7, we have $M = (P \oplus Q)$ with $H \vdash P\sigma : \phi$ (since $M\sigma = P\sigma \oplus Q\sigma$). By induction hypothesis there exists F such that $H \vdash_U \sigma : F$ where $U = \mathsf{fv}(P)$ and $F \vdash P : \phi$. If we let

$$G(x) = \begin{cases} F(x) & \text{if } x \in U \\ \omega & \text{otherwise} \end{cases}$$

then $H \vdash_V \sigma : G$ where $V = \mathsf{fv}(M)$ since $U \subseteq V$, and $G \vdash P : \phi$ by weakening, therefore $G \vdash M : \phi$ by L7. The argument is the same for L8 □

COROLLARY 5.8. *If $H \vdash M[N/x] : \phi$ then there exists ψ such that $H \vdash N : \psi$ and $x : \psi$, $H \vdash M : \phi$.*

PROOF: recall that $M[N/x]$ is $M\sigma$ where $\sigma(x) = N$, $\sigma(y) = y$ for $y \in \mathsf{fv}(M) - \{x\}$ and $\sigma(z) = \Omega$ for any other z. If $x \notin \mathsf{fv}(M)$ then we let $\psi = \omega$, and we have $H \vdash N : \psi$ by L1 and $x : \psi$, $H \vdash M : \phi$ by weakening. Otherwise by paste there exist G such that $H \vdash_V \sigma : G$ where $V = \mathsf{fv}(M)$ and $G \vdash M : \phi$. Therefore if $\psi = G(x)$ we have $H \vdash N : \psi$, and $H(y) \leq G(y)$ for $y \in V - \{x\}$ by the lemma 5.6. Then by weakening $x : \psi$, $H \vdash M : \phi$ □

We can now prove a first half of the validity of the β-law (and also the convergence lemma for closed terms). More precisely we show that reduction is decreasing with respect to "typing". To deal with the case of R6, we need to know something about the "typing" of the identity $\mathbf{I}$, and of terms $N \in \boldsymbol{N} - \boldsymbol{S}$.

LEMMA 5.9. *$H \vdash \mathbf{I} : \phi$ if and only if there exist formulae $\phi_1, \ldots, \phi_n$ such that*

$$(\phi_1 \to \phi_1) \wedge \cdots \wedge (\phi_n \to \phi_n) \leq \phi$$

PROOF: it is easy to see, using L4 and L5, that $H \vdash \mathbf{I} : \psi \to \psi$ for any ψ, hence $H \vdash \mathbf{I} : \phi$ if $(\phi_1 \to \phi_1) \wedge \cdots \wedge (\phi_n \to \phi_n) \leq \phi$ (using L2 and L3).

We prove the converse by induction on the proof of $H \vdash \mathbf{I} : \phi$. This is trivial if the last rule used to prove this sequent is L1 or (using the induction hypothesis) L3. For L2 we simply use the induction hypothesis and the remark 4.1. For L5 we have $\phi = \psi \to \zeta$ with $x : \psi$, $H \vdash x : \zeta$. Then by the lemma 5.6, $\psi \leq \zeta$, therefore $\psi \to \psi \leq \phi$ by E7 □

REMARK 5.10. $N \in \boldsymbol{N} - \boldsymbol{S} \Rightarrow N \vdash M : \omega \to \omega$.

This is true for $N = \lambda x P$, using L5 since $x:\omega$, $H \vdash P:\omega$ by L1. For $N = \nabla$, we use L9 and L3, since $(\omega \to \omega) \to (\phi \to \phi) \leq \omega \to \omega$ by E1, E7 and E6.

PROPOSITION (REDUCTION is DECREASING) 5.11. $M \stackrel{*}{\Rightarrow} M' \;\&\; H \vdash M':\phi \;\Rightarrow\; H \vdash M:\phi$

PROOF: clearly it is enough to prove this fact for $M \Rightarrow M'$. We proceed by induction on the proof of this reduction:

• if $M \Rightarrow M'$ is proved using R1 then $M = (\lambda x N)R$ and $M' = N[R/x]$. Then by paste (corollary 5.8) there exists ψ such that $H \vdash R:\psi$ and $x:\psi$, $H \vdash N:\phi$. Therefore by L5 we have $H \vdash \lambda x N:\psi \to \phi$, and $H \vdash M:\phi$ by L6.

• by R2 we have $M = (NP)$ and $M' = (N'P)$ with $N \Rightarrow N'$. We proceed by induction of the proof of the sequent $H \vdash M':\phi$. If this is an instance of L1, $\phi = \omega$ and we obviously have also $H \vdash M:\phi$ by L1. If the last rule used to prove this sequent is L2 or L3, we simply use the induction hypothesis. If $H \vdash M':\phi$ is proved using L6 then $H \vdash N':\psi \to \phi$ and $H \vdash P:\psi$ for some ψ. Then we use the induction hypothesis (on the reduction $N \Rightarrow N'$) and L6.

• by R3 we have $M = (M' \oplus N)$, therefore $H \vdash M:\phi$ by L7. Similarly for R4 we use L8.

• by R5 we have $M = \nabla N$ and $M' = \mathbf{I}$ with $N \in \boldsymbol{N} - \boldsymbol{S}$. By the lemma above we may assume (using L2 and L3) that $\phi = \psi \to \psi$ for some ψ. Then by the preceding remark and L9, L6 we have $H \vdash M:\psi \to \psi$.

• by R6, $M = \nabla N$ and $M' = \nabla N'$ with $N \Rightarrow N'$. We proceed by induction of the proof of the sequent $H \vdash M':\phi$. If this is an instance of L1, $\phi = \omega$ and we obviously have also $H \vdash M:\phi$ by L1. If the last rule used to prove this sequent is L2 or L3, we simply use the induction hypothesis. If $H \vdash M':\phi$ is proved using L6 then $H \vdash \nabla:\psi \to \phi$ and $H \vdash N':\psi$ for some ψ. Then we use the induction hypothesis (on the reduction $N \Rightarrow N'$) and L6 □

An obvious consequence of these results is that the convergence lemma for closed terms holds, as well as a half of the validity of the β-law:

COROLLARY 5.12.

(i) $M[N/x] \sqsubseteq_{\mathcal{S}} (\lambda x M)N$

(ii) *for* M *closed* $M{\Downarrow} \;\Rightarrow\; \vdash M:\omega \to \omega$.

5.3 The Restricted Sequent Calculus.

To establish the remaining results (second half of the validity of the β-law and the characterization lemma) it will be convenient to use a sequent calculus which, although incomplete, allows us to prove the "essential" formulae about the terms. This restricted sequent calculus is quite close to the original Coppo's system [12,13]. The (provable) sequents in this system will be denoted $H \Vdash M:\phi$, though the restriction actually concerns the formulae.

To see what are the special formulae we use, let us define inductively the *decomposition* relation $\phi \triangleright \psi$ between formulae, as follows:

(i) $\omega \triangleright \omega$

(ii) $\phi \triangleright \phi' \;\Rightarrow\; \begin{cases} \phi \wedge \psi \triangleright \phi' \\ \psi \wedge \phi \triangleright \phi' \\ \psi \to \phi \triangleright \psi \to \phi' \end{cases}$

REMARK 5.13. $\phi \triangleright \psi \;\Rightarrow\; \phi \leq \psi$

DEFINITION (IRREDUCIBLE FORMULAE). *A formula* ϕ *is irreducible if* $\phi \triangleright \psi \;\Rightarrow\; \psi = \phi$.

It is easily seen that a formula is irreducible if and only if it is ω or has the form $(\psi \to \phi)$ where ϕ is irreducible. Moreover it is easy to see that if $\phi \triangleright \psi$ then ψ is an irreducible formula. We shall denote by Ψ the set of irreducible formulae. In what follows we study the entailment between formulae by means of irreducible ones (*cf.* Barendregt & al. [7], Hindley 82[24], Abramsky 88[2]).

LEMMA 5.14. *For any ϕ the set $\phi^* = \{\psi \mid \phi \triangleright \psi\}$ is non-empty and finite. Moreover $\phi \sim \bigwedge\{\psi \mid \phi \triangleright \psi\}$.*

PROOF (sketch): by induction on the definition of $\phi \triangleright \psi$, trivial. Note that $(\phi \wedge \psi)^* = \phi^* \cup \psi^*$. For the case $\psi \to \phi$ one uses the equivalence $(\psi \to \bigwedge_{1 \le i \le n} \phi_i) \sim \bigwedge_{1 \le i \le n} (\psi \to \phi_i)$ ❑

A consequence of this fact is that any irreducible formula (distinct from ω) can be written, up to $\sim$, as $\psi_1 \wedge \cdots \wedge \psi_k \to \psi$ where ψ and the ψ_i's are irreducible. Let $\ll$ be the relation on Ψ given inductively by

$$\boxed{\text{F1}: \ \phi \ll \omega \qquad \text{F2}: \ \phi \ll \phi}$$

$$\boxed{\text{F3}: \ \frac{\phi \ll \tau \ , \ \tau \ll \psi}{\phi \ll \psi} \qquad \text{F4}: \ (\phi \to \omega) \ll (\omega \to \omega)}$$

$$\boxed{\text{F5}: \ \frac{\phi_0 \ge \phi_1 \ , \ \psi_0 \ll \psi_1}{(\phi_0 \to \psi_0) \ll (\phi_1 \to \psi_1)}}$$

LEMMA 5.15. $\phi \le \psi \ \Leftrightarrow \ \forall \psi' \ (\psi \triangleright \psi' \ \Rightarrow \ \exists \phi' \ (\phi \triangleright \phi' \ \& \ \phi' \ll \psi'))$

PROOF: the "$\Leftarrow$" part is clear since for any formula ζ we have $\zeta \sim \bigwedge\{\zeta' \mid \zeta \triangleright \zeta'\}$. The converse implication is easily established by induction on the definition of $\phi \le \psi$ ❑

We could have written this property as:

$$\phi \le \psi \ \Leftrightarrow \ \forall \psi' \in \psi^* \ \exists \phi' \in \phi^* . \ \phi' \ll \psi'$$

This shows that the canonical domain $\boldsymbol{D}_*$ is the *lower powerdomain* (i.e. Hoare powerdomain) generated by its compact prime elements, that is the filters generated by the irreducible formulae.

LEMMA 5.16. *If $\tau \le (\phi \to \psi)$ then*

$$\bigwedge\{\zeta \mid \exists \gamma \ge \phi . \ (\gamma \to \zeta) \in \tau^*\} \le \psi$$

PROOF: let $f = \{\zeta \mid \exists \gamma \ge \phi . \ (\gamma \to \zeta) \in \tau^*\}$. Since $\phi \to \psi \sim (\phi \to \bigwedge_{\psi \triangleright \psi'} \psi') \sim \bigwedge_{\psi \triangleright \psi'} (\phi \to \psi')$ we have $\tau \le \phi \to \psi$ if and only if $\tau \le \phi \to \psi'$ for any ψ' such that $\psi \triangleright \psi'$. Similarly $\bigwedge_{\zeta \in f} \xi \le \psi$ if and only if $\bigwedge_{\zeta \in f} \xi \le \psi'$ for any ψ' such that $\psi \triangleright \psi'$. Therefore we may assume that ψ is irreducible. By the previous lemma there exists τ' such that $\tau \triangleright \tau'$ and $\tau' \ll \phi \to \psi$. It is easy prove, by induction on the inference of $\tau' \ll \phi \to \psi$, that either $\psi = \omega$ or $\tau' = \gamma \to \zeta$ with $\gamma \ge \phi$ and $\zeta \ll \psi$. In the first case the result is trivial, by E2, and in the second one $\zeta \in f$, hence $\bigwedge_{\zeta \in f} \xi \le \psi$ ❑

COROLLARY 5.17. $\phi_0 \to \psi_0 \le \phi_1 \to \psi_1 \ \Leftrightarrow \ \psi_1 \sim \omega \quad \text{or} \quad \phi_0 \ge \phi_1 \ \& \ \psi_0 \le \psi_1$

PROOF: by the lemma we have

$$\bigwedge\{\zeta \mid \exists \gamma \ge \phi_1 . \ (\gamma \to \zeta) \in (\phi_0 \to \psi_0)^*\} \le \psi_1$$

If $\{\zeta \mid \exists \gamma \geq \phi_1.\ (\gamma \to \zeta) \in (\phi_0 \to \psi_0)^\star\} = \emptyset$ then $\psi_1 \geq \omega$ (the empty conjunction is ω). Otherwise there exist $\gamma \geq \phi_1$ and ζ such that $\gamma \to \zeta \in (\phi_0 \to \psi_0)^\star$. Since

$$\xi \in (\phi_0 \to \psi_0)^\star \ \Leftrightarrow\ \exists \zeta.\ \psi_0 \triangleright \zeta\ \&\ \xi = \phi_0 \to \zeta$$

we have $\gamma = \phi_0$, hence $\phi_0 \geq \phi_1$. Moreover $\{\zeta \mid \exists \gamma \geq \phi_1.\ (\gamma \to \zeta) \in (\phi_0 \to \psi_0)^\star\} = \psi_0^*$, therefore by the previous lemma $\psi_0 \leq \psi_1$ ❑

For instance the formulae $(\omega \to (\omega \to \omega)) \to (\omega \to (\omega \to \omega))$ and $(\omega \to \omega) \to (\omega \to \omega)$ are incomparable, since

$$\omega \to (\omega \to \omega) < (\omega \to \omega) \to (\omega \to \omega) < \omega \to \omega < \omega$$

Now we introduce the restricted sequent calculus, which essentially consists in disallowing L2 and L3.

$$\text{T1}: \quad H \Vdash M : \omega$$

$$\text{T2}: \quad \frac{}{x : \phi,\ H \Vdash x : \psi}\ \phi \triangleright \psi \qquad\qquad \text{T3}: \quad \frac{x : \phi,\ H \Vdash M : \psi}{H \Vdash \lambda x M : \phi \to \psi}$$

$$\text{T4}: \quad \frac{H \Vdash M : (\phi \to \psi)\ ,\ H \Vdash N : \phi_i \quad (1 \leq i \leq k)}{H \Vdash (M\,N) : \psi}\ \phi_1 \wedge \cdots \wedge \phi_k \leq \phi$$

$$\text{T5}: \quad \frac{H \Vdash M : \phi}{H \Vdash (M \oplus N) : \phi} \qquad\qquad \text{T6}: \quad \frac{H \Vdash N : \psi}{H \Vdash (M \oplus N) : \psi}$$

$$\text{T7}: \quad \frac{}{H \Vdash \nabla : (\omega \to \omega) \to (\zeta \to \tau)}\ \zeta \triangleright \tau$$

It is easily checked, by induction on the inference of the sequents, that:

REMARK. *If $H \Vdash M : \phi$ then ϕ is an irreducible formula.*

REMARK 5.18. $x : \phi,\ H \Vdash x : \psi \ \Rightarrow\ \phi \leq \psi$

Now we relate the two logical systems:

PROPOSITION 5.19.

(i) $H \Vdash M : \phi \ \Rightarrow\ H \vdash M : \phi$

(ii) $H \vdash M : \phi \ \Rightarrow\ \exists \psi \leq \phi\ \forall \psi'.\ \psi \triangleright \psi' \ \Rightarrow\ H \Vdash M : \psi'$

PROOF: one easily proves the first point by induction on the inference of the sequent $H \Vdash M : \phi$, using the remark 5.18 above. Let us just check the case of T4: we have $H \Vdash P : \phi \to \psi$ and $H \Vdash Q : \phi_i$ with $\phi_1 \wedge \cdots \wedge \phi_k \leq \phi$. Then by induction hypothesis $H \vdash P : \phi \to \psi$ and $H \vdash Q : \phi_i$, therefore $H \vdash Q : \phi$ by L2 and L3, hence $H \vdash (PQ) : \psi$ by L6.

Now we prove the second point by induction on the inference of the sequent $H \vdash M : \phi$. This is trivial for L1, L4 and L3 (using the induction hypothesis).

• for L2 we have $\phi = \phi_0 \wedge \phi_1$ with $H \vdash M : \phi_i$ $(i = 0, 1)$. Then by induction hypothesis there exist ψ_0 and ψ_1 such that $\psi_i \leq \phi_i$ and $H \Vdash M : \psi_i'$ for any ψ_i' such that $\psi_i \triangleright \psi_i'$. It is easy to see that $\psi = \psi_0 \wedge \psi_1$ fulfils the requirements of the proposition.

• for L5 we have $\phi = \xi \to \tau$ and $M = \lambda x N$ with $x : \xi$, $H \vdash N : \tau$. By induction hypothesis there exists ζ such that $\zeta \leq \tau$ and $x : \xi$, $H \Vdash N : \zeta'$ for each ζ' such that $\zeta \triangleright \zeta'$. If we let $\psi = \xi \to \zeta$ then $\psi \leq \phi$ by E7. If $\psi \triangleright \psi'$ then there exists ζ' such that $\zeta \triangleright \zeta'$ and $\psi' = \xi \to \zeta'$, hence $H \Vdash M : \psi'$ by T3.

• for L6 we have $M = (PQ)$ and $H \vdash P : \phi' \to \phi$ and $H \vdash Q : \phi'$. By induction hypothesis, there exist τ and γ such that $\tau \leq (\phi' \to \phi)$, $\gamma \leq \phi'$, $\tau \triangleright \tau' \Rightarrow H \Vdash P : \tau'$ and $\gamma \triangleright \gamma' \Rightarrow H \Vdash Q : \gamma'$. Let

$$\psi = \bigwedge \{\zeta \mid \exists \zeta' \geq \phi' .\ (\zeta' \to \zeta) \in \tau^*\}$$

Then $\psi \leq \phi$ by the lemma 5.16. Clearly $\psi \triangleright \psi'$ if and only if there exists $\zeta' \geq \phi'$ such that $\tau \triangleright (\zeta' \to \psi')$. Then $H \Vdash P : (\zeta' \to \psi')$, and we may use T4 to conclude $H \Vdash M : \psi'$ since $\gamma \sim \bigwedge \{\gamma' \mid \gamma \triangleright \gamma'\} \leq \zeta'$ and $H \Vdash Q : \gamma'$ for any $\gamma' \in \gamma^*$.

• the cases of L7 and L8 are trivial.

• for L9 we have $M = \nabla$ and $\phi = (\omega \to \omega) \to (\tau \to \tau)$. We let $\psi = \phi$; clearly if $\psi \triangleright \psi'$ then $\psi' = (\omega \to \omega) \to (\tau \to \tau')$ with $\tau \triangleright \tau'$, therefore $H \Vdash \nabla : \psi'$ by T7 □

Using this result it is now easy to prove that the logical interpretation satisfies the β-law:

LEMMA 5.20. $H \vdash (\lambda x M)N : \phi \Rightarrow H \vdash M[N/x] : \phi$

PROOF: by the previous proposition if $H \vdash (\lambda x M)N : \phi$ then there exists $\psi \leq \phi$ such that

$$\psi \triangleright \psi' \Rightarrow H \Vdash (\lambda x M)N : \psi'$$

Let us show by induction on the proof of this last sequent that $H \vdash M[N/x] : \psi'$. In fact there are only two cases, T1, which is trivial, and T4. In this case there exist ζ and $\psi_1, \ldots, \psi_k$ such that $\psi_1 \wedge \cdots \wedge \psi_k \leq \zeta$,

(1) $$H \Vdash N : \psi_i \quad \text{for all } i$$

and $H \Vdash \lambda x M : \zeta \to \psi'$. This last sequent cannot be proved using T1, therefore it is proved by means of T3, that is:

(2) $$x : \zeta\ ,\ H \Vdash M : \psi'$$

Then by the previous proposition we get from (2)

$$x : \zeta\ ,\ H \vdash M : \psi'$$

and from (1)

$$H \vdash N : \psi_i \quad \text{for all } i$$

therefore $H \vdash N : \psi_1 \wedge \cdots \wedge \psi_k$ by L2, hence $H \vdash N : \zeta$ by L3. By cut (corollary 5.4) we have $H \vdash M[N/x] : \psi'$.

Since this holds for any ψ' such that $\psi \triangleright \psi'$, by the lemma 5.14 and L2, L3 we then have $H \vdash M[N/x] : \psi$, hence $H \vdash M[N/x] : \phi$ by L3 □

LEMMA 5.21.

(i) $H \Vdash \mathbf{I} : \zeta \;\Rightarrow\; \exists \phi.\, \phi \to \phi \leq \zeta$

(ii) $H \Vdash \mathbf{T} : \zeta \;\Rightarrow\; \exists \phi.\, \phi \to (\omega \to \phi) \leq \zeta$

(iii) $H \Vdash \mathbf{F} : \zeta \;\Rightarrow\; \exists \phi.\, \omega \to (\phi \to \phi) \leq \zeta$

The proof is left as an exercise.

5.4 Characteristic Terms.

DEFINITION (TRIVIAL FORMULAE). *A formula ϕ is trivial if $\phi \sim \omega$.*

Clearly ϕ is trivial if and only if $\omega \leq \phi$. Then $\phi \wedge \psi$ is trivial if and only if both ϕ and ψ are trivial.

LEMMA 5.22.

(i) $\phi \sim \omega \;\Leftrightarrow\; \phi^* = \{\omega\}$

(ii) $\phi \not\sim \omega \;\Leftrightarrow\; \phi \leq \omega \to \omega$

PROOF:

(i) if $\phi \sim \omega$ then $\omega \leq \phi$ and by the lemma 5.15 there exists $\psi \in \phi^*$ such that $\omega \ll \psi$. It is easy to see, by induction on the definition of $\ll$, that $\omega \ll \psi \;\Leftrightarrow\; \psi = \omega$. Therefore $\phi^* = \{\omega\}$.

(ii) since $(\omega \to \omega)^* = \{\omega \to \omega\}$, the formula $\omega \to \omega$ is non-trivial, by the previous point, and clearly $\phi \leq \psi \not\sim \omega \;\Rightarrow\; \phi \not\sim \omega$. Conversely if $\phi \not\sim \omega$ then by the lemma 5.15 there exists $\psi \in \phi^*$ such that $\omega \not\ll \psi$. It is obvious that this implies $\psi = \psi_0 \to \psi_1$ since ψ is irreducible, hence $\psi \leq \omega \to \omega$ by E1, E7 and E6. Then $\phi \leq \omega \to \omega$ since $\phi \leq \psi$ (remark 5.13) □

We define for each formula $\phi \in \Phi$ a pair of closed terms $\mathbf{M}_\phi$ and $\mathbf{T}_\phi$ of $\Lambda_J(\nabla)$ such that:

(i) $\vdash \mathbf{M}_\phi : \phi$

(ii) $\vdash \mathbf{T}_\phi : \phi \to (\zeta \to \zeta)$

We define these terms inductively, checking the required properties:

$$\boxed{\begin{array}{l} \mathbf{M}_\omega = \Omega \\ \mathbf{T}_\omega = \mathbf{F} = \lambda y \lambda x x \end{array}}$$

Clearly $\vdash \mathbf{M}_\omega : \omega$ by L1. To show that $\vdash \mathbf{F} : \omega \to (\zeta \to \zeta)$ is an easy exercise.

$$\boxed{\begin{array}{l} \mathbf{M}_{\phi\wedge\psi} = (\mathbf{M}_\phi \oplus \mathbf{M}_\psi) \\ \mathbf{T}_{\phi\wedge\psi} = \lambda x (\mathbf{T}_\phi x)(\mathbf{T}_\psi x) \end{array}}$$

By induction we have $\vdash \mathbf{M}_\phi : \phi$ and $\vdash \mathbf{M}_\psi : \psi$, therefore by L7, L8 and L2:

$$\vdash (\mathbf{M}_\phi \oplus \mathbf{M}_\psi) : \phi \wedge \psi$$

as required. By induction, for any τ we have $\vdash \mathbf{T}_\phi : \phi \to (\tau \to \tau)$, hence by weakening

$$x : \phi \wedge \psi \vdash \mathbf{T}_\phi : \phi \to (\tau \to \tau)$$

Moreover by L4 and L3: $x:\phi\wedge\psi\vdash x:\phi$. Then using L6:

$$x:\phi\wedge\psi\vdash \mathbf{T}_\phi x:\tau\to\tau$$

Similarly we have $x:\phi\wedge\psi\vdash \mathbf{T}_\psi x:\zeta\to\zeta$. Let $\tau=(\zeta\to\zeta)$; then by L6:

$$x:\phi\wedge\psi\vdash(\mathbf{T}_\phi x)(\mathbf{T}_\psi x):\zeta\to\zeta$$

and finally $\vdash\lambda x(\mathbf{T}_\phi x)(\mathbf{T}_\psi x):(\phi\wedge\psi)\to(\zeta\to\zeta)$ by L5, that is the required property of $\mathbf{T}_{\phi\wedge\psi}$.

$$\boxed{\begin{array}{l}\mathbf{M}_{\phi\to\psi}=\lambda x(\mathbf{T}_\phi x)\mathbf{M}_\psi\\ \mathbf{T}_{\phi\to\psi}=\lambda x(\nabla x)(\mathbf{T}_\psi(x\mathbf{M}_\phi))\end{array}}$$

By induction $x:\phi\vdash\mathbf{M}_\psi:\psi$. As we saw above

$$x:\phi\vdash\mathbf{T}_\phi x:\psi\to\psi$$

Then by L6:

$$x:\phi\vdash(\mathbf{T}_\phi x)\mathbf{M}_\psi:\psi$$

and by L5 we get the required property of $\mathbf{M}_{\phi\to\psi}$, namely:

$$\vdash\lambda x(\mathbf{T}_\phi x)\mathbf{M}_\psi:\phi\to\psi$$

By L9 we have $x:\phi\to\psi\vdash\nabla:(\omega\to\omega)\to(\tau\to\tau)$ for any τ. Then

$$x:\phi\to\psi\vdash\nabla:(\phi\to\psi)\to(\tau\to\tau)$$

by L3 and E7, since, by E1, E7 and E6:

$$\phi\to\psi\le\phi\to\omega\le\omega\to\omega$$

Since we also have $x:\phi\to\psi\vdash x:\phi\to\psi$ by L4, we get by L6:

(1) $$x:\phi\to\psi\vdash\nabla x:\tau\to\tau$$

From $\vdash\mathbf{M}_\phi:\phi$ it is easy to get by weakening, L4 and L6:

$$x:\phi\to\psi\vdash x\mathbf{M}_\phi:\psi$$

Then similarly from $x:\phi\to\psi\vdash\mathbf{T}_\psi:\psi\to(\zeta\to\zeta)$, we have:

(2) $$x:\phi\to\psi\vdash\mathbf{T}_\psi(x\mathbf{M}_\phi):\zeta\to\zeta$$

If we let $\tau=\zeta\to\zeta$ we now have, from (1) and (2) by L6:

$$x:\phi\to\psi\vdash(\nabla x)(\mathbf{T}_\psi(x\mathbf{M}_\phi)):\zeta\to\zeta$$

thus finally $\vdash\lambda x(\nabla x)(\mathbf{T}_\psi(x\mathbf{M}_\phi)):(\phi\to\psi)\to(\zeta\to\zeta)$ by L5.

PROPOSITION 5.23.

(i) $H\Vdash\mathbf{M}_\phi:\psi\ \Rightarrow\ \phi\le\psi$

(ii) $H\Vdash\mathbf{T}_\phi:\psi\to(\zeta\to\tau)\ \Rightarrow\ \phi\ge\psi\ \&\ \zeta\le\tau$.

PROOF: by induction on the formula ϕ.

• assume that $H \Vdash \mathbf{M}_\omega : \psi$, that is $H \Vdash \Omega : \psi$, with $\psi \not\sim \omega$. Then we would have by the lemma 5.22, $\psi \leq \omega \to \omega$, hence $H \vdash \Omega : \omega \to \omega$ by the proposition 5.19 and L3. But then by soundness we would have $\models \Omega : \omega \to \omega$, and this is impossible since $\Omega\Uparrow$. Therefore $H \Vdash \mathbf{M}_\omega : \psi \;\Rightarrow\; \omega \leq \psi$.

• if $H \Vdash \mathbf{T}_\omega : \psi \to (\zeta \to \tau)$ then by the lemma 5.21, $\omega \to (\gamma \to \gamma) \leq \psi \to (\zeta \to \tau)$ for some γ (since $\mathbf{T}_\omega = \mathbf{F}$). By the corollary 5.17, $\psi \geq \omega$ and $\gamma \to \gamma \leq \zeta \to \tau$ (since $\zeta \to \tau \not\sim \omega$, by the lemma 5.22), hence by the same corollary either $\tau \sim \omega$, in which case $\zeta \leq \tau$, or $\gamma \geq \zeta$ and $\gamma \leq \tau$, which implies $\zeta \leq \tau$.

• if $H \Vdash (\mathbf{M}_\phi \oplus \mathbf{M}_\psi) : \zeta$ then this sequent is proved either by means of T1, in which case $\zeta = \omega$ (hence $\phi \wedge \psi \leq \zeta$), or using T5 or T6. Then by induction hypothesis $\phi \leq \zeta$ or $\psi \leq \zeta$, hence $\phi \wedge \psi \leq \zeta$ in any case.

• the sequent $H \Vdash \mathbf{T}_{\phi\wedge\psi} : \gamma \to (\zeta \to \tau)$ can only be proved by means of T3, that is:

$$x : \gamma \,,\, H \Vdash (\mathbf{T}_\phi x)(\mathbf{T}_\psi x) : \zeta \to \tau$$

Since this one must be proved using T4, there exist $\nu_1, \ldots, \nu_m$ such that $\nu_1 \wedge \cdots \wedge \nu_m \leq \nu$ and

(1-i) $$x : \gamma \,,\, H \Vdash \mathbf{T}_\psi x : \nu_i$$

(2) $$x : \gamma \,,\, H \Vdash \mathbf{T}_\phi x : \nu \to (\zeta \to \tau)$$

This second sequent is proved by T4, thus:

(3) $$x : \gamma \,,\, H \Vdash x : \mu_j \qquad \text{with} \quad \mu_1 \wedge \cdots \wedge \mu_n \leq \mu$$

(4) $$x : \gamma \,,\, H \Vdash \mathbf{T}_\phi : \mu \to (\nu \to (\zeta \to \tau))$$

Then $\gamma \leq \mu$ by the remark 5.18. Moreover $\mu \leq \phi$ by induction hypothesis regarding (4), hence $\gamma \leq \phi$ (recall that we have to show $\phi \wedge \psi \geq \gamma$ and $\zeta \leq \tau$), and $\nu \leq \zeta \to \tau$ (induction hypothesis), hence $\nu_1 \wedge \cdots \wedge \nu_m \leq \zeta \to \tau$. Then, since each ν_i is irreducible, from the lemma 5.16 there exists i, ξ and δ such that $\nu_i = \xi \to \delta$ with $\xi \geq \zeta$ and $\delta \leq \tau$. Therefore the sequent (1-i) is proved using T4, that is

$$x : \gamma \,,\, H \Vdash x : \eta_h \qquad \text{with} \quad \eta_1 \wedge \cdots \wedge \eta_r \leq \eta$$
$$x : \gamma \,,\, H \Vdash \mathbf{T}_\psi : \eta \to (\xi \to \delta)$$

By induction hypothesis $\eta \leq \psi$, and by the remark 5.18, $\gamma \leq \eta_h$, hence

$$\gamma \leq \eta_1 \wedge \cdots \wedge \eta_r \leq \eta \leq \psi$$

and finally $\gamma \leq \phi \wedge \psi$. Moreover, by induction hypothesis $\xi \leq \delta$, therefore $\zeta \leq \xi \leq \delta \leq \tau$, that is $\zeta \leq \tau$.

• if $H \Vdash \mathbf{M}_{\phi\to\psi} : \zeta$, then this sequent is proved either by means of T1, in which case $\zeta = \omega$ (hence $\phi \to \psi \leq \zeta$), or using T3. In this case $\zeta = \xi \to \tau$ with

$$x : \xi \,,\, H \Vdash (\mathbf{T}_\phi x)\mathbf{M}_\psi : \tau$$

If this sequent is proved using T1 then $\tau = \omega$ and we have $\phi \to \psi \leq \zeta$ since:

$$\phi \to \psi \leq \phi \to \omega \leq \omega \to \omega \leq \xi \to \omega$$

Otherwise there exist $\nu_1, \ldots, \nu_m$ such that $\nu_1 \wedge \cdots \wedge \nu_m \leq \nu$ and

$$x : \xi\,,\ H \Vdash \mathbf{M}_\psi : \nu_i$$
$$x : \xi\,,\ H \Vdash \mathbf{T}_\phi x : \nu \to \tau$$

By induction hypothesis $\psi \leq \nu_i$ for all i, therefore $\psi \leq \nu$. The second sequent is proved by T4, thus:

$$x : \xi\,,\ H \Vdash x : \mu_j \qquad \text{with} \quad \mu_1 \wedge \cdots \wedge \mu_n \leq \mu$$
$$x : \xi\,,\ H \Vdash \mathbf{T}_\phi : \mu \to (\nu \to \tau)$$

Then $\xi \leq \mu_j$ by the remark 5.18, hence $\xi \leq \mu$. Moreover $\mu \leq \phi$ and $\nu \leq \tau$ by induction hypothesis, hence $\xi \leq \phi$ and $\psi \leq \tau$, which implies $\phi \to \psi \leq \xi \to \tau = \zeta$.

- the sequent $H \Vdash \mathbf{T}_{\phi \to \psi} : \gamma \to (\zeta \to \tau)$ can only be proved by means of T3, that is:

$$x : \gamma\,,\ H \Vdash (\nabla x)(\mathbf{T}_\psi(x\mathbf{M}_\phi)) : \zeta \to \tau$$

Since this one must be proved using T4, there exist $\delta_1, \ldots, \delta_k$ such that $\delta_1 \wedge \cdots \wedge \delta_k \leq \delta$ and

(1-i) $$x : \gamma\,,\ H \Vdash \mathbf{T}_\psi(x\mathbf{M}_\phi) : \delta_i$$

(2) $$x : \gamma\,,\ H \Vdash \nabla x : \delta \to (\zeta \to \tau)$$

This second sequent is proved by T4, thus:

(3) $$x : \gamma\,,\ H \Vdash x : \mu_j \qquad \text{with} \quad \mu_1 \wedge \cdots \wedge \mu_p \leq \mu$$

(4) $$x : \gamma\,,\ H \Vdash \nabla : \mu \to (\delta \to (\zeta \to \tau))$$

This last sequent is proved by means of T7, therefore $\delta \triangleright \zeta \to \tau$, hence $\delta_1 \wedge \cdots \wedge \delta_k \leq \zeta \to \tau$. Then, since each δ_i is irreducible, from the lemma 5.16 there exists i, ξ and κ such that $\delta_i = \xi \to \kappa$ with $\xi \geq \zeta$ and $\kappa \leq \tau$. Therefore the sequent (1-i) is proved using T4, that is

(5-h) $$x : \gamma\,,\ H \Vdash x\mathbf{M}_\phi : \eta_h \qquad \eta_1 \wedge \cdots \wedge \eta_r \leq \eta$$

(6) $$x : \gamma\,,\ H \Vdash \mathbf{T}_\psi : \eta \to (\xi \to \kappa)$$

By induction hypothesis regarding (6) we have $\xi \leq \kappa$, hence $\zeta \leq \tau$ (since $\xi \geq \zeta$ and $\kappa \leq \tau$). It remains to show $\phi \to \psi \geq \gamma$. Note that by induction hypothesis for (6) we have $\psi \geq \eta$, hence $\eta_1 \wedge \cdots \wedge \eta_r \leq \psi$. The sequents (5-$h$) are proved using either T1 or T4. In the first case $\eta_h = \omega$, and using the remark 5.18 we get from (3) $\gamma \leq \mu_j$ for any j, hence $\gamma \leq \mu$. Moreover the sequent (4) is proved by means of T7, therefore $\mu = \omega \to \omega$, hence $\gamma \leq \omega \to \omega$. Since $\omega \to \omega \leq \phi \to \eta_h$, we have $\gamma \leq \phi \to \eta_h$ in this case (T1 for 5-h). Otherwise (T4):

$$x : \gamma\,,\ H \Vdash x : \varepsilon \to \eta_h$$
$$x : \gamma\,,\ H \Vdash \mathbf{M}_\phi : \varepsilon_j \qquad \text{with} \quad \varepsilon_1 \wedge \cdots \wedge \varepsilon_s \leq \varepsilon$$

By the remark 5.18, $\gamma \leq \varepsilon \to \eta_h$, hence $\gamma \leq \phi \to \eta_h$ since by induction hypothesis $\phi \leq \varepsilon_j$ for any j. Then we have

$$\gamma \leq \bigwedge_h (\phi \to \eta_h) \sim \phi \to (\eta_1 \wedge \cdots \wedge \eta_s) \leq \phi \to \eta$$

and finally $\gamma \leq \phi \to \psi$ since $\eta \leq \psi$ by induction hypothesis regarding (6) □

COROLLARY (CHARACTERIZATION LEMMA) 5.24. *For any formula ϕ*

$$\vdash \mathbf{M}_\phi : \psi \;\Leftrightarrow\; \phi \leq \psi$$

PROOF: we have seen that $\vdash \mathbf{M}_\phi : \phi$, therefore if $\phi \leq \psi$ we have $\vdash \mathbf{M}_\phi : \psi$ by L3. Conversely if $\vdash \mathbf{M}_\phi : \psi$ then by the proposition 5.19 there exist $\psi_1, \ldots, \psi_k$ such that

$$\psi_1 \wedge \cdots \wedge \psi_k \leq \psi \quad \text{and} \quad \forall i.\ \Vdash \mathbf{M}_\phi : \psi_i$$

Then we have $\phi \leq \psi_i$ for all i by the previous proposition, hence $\phi \leq \psi$ by E5 □

REFERENCES

[1] S. ABRAMSKY, *Domain theory in logical form*, LICS 87 (1987) 47-53.

[2] S. ABRAMSKY, *The lazy lambda-calculus*, in Declarative Programming, Ed. D. Turner, Addison Wesley (1989).

[3] S. ABRAMSKY, C.-H. LUKE ONG, *Full abstraction in the lazy lambda-calculus*, Research Report, Dept. of Computing, Imperial College (1989).

[4] L. ACETO, M. HENNESSY, *Termination, deadlock and divergence*, Comput. Sci. Report 6/88, University of Sussex (1988).

[5] E. A. ASHCROFT, M. HENNESSY, *A mathematical semantics for a non-deterministic typed λ-calculus*, Theoretical Comput. Sci. 11 (1980) 227-245.

[6] H. BARENDREGT, *The Lambda Calculus*, Studies in Logic 103, North-Holland, Revised Edition (1984).

[7] H. BARENDREGT, M. COPPO, M. DEZANI-CIANCAGLINI, *A filter lambda model and the completeness of type assignment*, J. of Symbolic Logic 48 (1983) 931-940.

[8] G. BERRY, P.-L. CURIEN, J.-J. LÉVY, *Full abstraction for sequential languages: the state of the art*, in Algebraic Methods in Semantics (M. Nivat & J. C. Reynolds, Eds), Cambridge University Press (1985) 90-132.

[9] G. BERRY, G. BOUDOL, *The chemical abstract machine*, POPL 90 (1990) 81-94.

[10] B. BLOOM, *Can LCF be topped? Flat lattice models of typed lambda-calculus*, LICS 88 (1988) 282-295.

[11] G. BOUDOL, *Towards a lambda-calculus for concurrent and communicating systems*, TAPSOFT 89, Lecture Notes in Comput. Sci. 351 (1989) 149-161.

[12] M. COPPO, M. DEZANI-CIANCAGLINI, B. VENERI, *Functional characters of solvable terms*, Zeit. Math. Logik Grund. 27 (1981) 45-58.

[13] M. COPPO, M. DEZANI-CIANCAGLINI, B. VENERI, *Principal type schemes and lambda-calculus semantics*, In To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism (J.R. Hindley and J.P. Seldin, Eds.), Academic Press (1980) 535-560.

[14] M. COPPO, M. DEZANI-CIANCAGLINI, F. HONSELL, G. LONGO, *Extended type structures and filter lambda models*, Logic Colloquium 82, North-Holland (1984) 241-262.

[15] M. COPPO, M. DEZANI-CIANCAGLINI, G. LONGO, *Applicative information systems*, CAAP 83, Lecture Notes in Comput. Sci. 159 (1983) 35-64.

[16] M. COPPO, *Completeness of type assignment in continuous lambda models*, Theoretical Comput. Sci. 29 (1984) 309-324.

[17] P.-L. CURIEN, *Categorical Combinators, Sequential Algorithms and Functional Programming*, Research Notes in Theoretical Computer Science, Pitman, Wiley (1986).

[18] Ph. DARONDEAU, *About fair asynchrony*, Theoretical Comput. Sci. 37 (1985) 305-336.

[19] R. DE NICOLA, M. HENNESSY, *Testing equivalences for processes*, Theoretical Comput. Sci. 34 (1984) 83-133.

[20] M. DEZANI-CIANCAGLINI, I. MARGARIA, *A characterization of F-complete type assignments*, Theoretical Comput. Sci. 45 (1986) 121-157.

[21] M. HENNESSY, G. PLOTKIN, *Full abstraction for a simple parallel programming language*, MFCS 79, Lecture Notes in Comput. Sci. 74 (1979) 108-120.

[22] M. HENNESSY, G. PLOTKIN, *A term model for* CCS, MFCS 80, Lecture Notes in Comput. Sci. 88 (1980) 261-274.

[23] R. HINDLEY, *The completeness theorem for typing λ-terms*, Theoretical Comput. Sci. 22 (1983) 1-17 and 127-133.

[24] R. HINDLEY, *The simple semantics for Coppo-Dezani-Sallé types*, Intern. Symp. on Programming, Lecture Notes in Comput. Sci. 137 (1982) 212-226.

[25] R. JAGADEESAN, P. PANANGADEN, *A domain theoretic model for a higher-order process calculus*, Comput. Sci. Department, Cornell University (1989).

[26] J.-L. KRIVINE, *Les lambda-calculs typés*, Cours de DEA, Université Paris 7 (1987-88).

[27] D. LEIVANT, *Typing and computational properties of lambda expressions*, Theoretical Comput. Sci. 44 (1986) 51-68.

[28] J.-J. LÉVY, *An algebraic interpretation of the $\lambda\beta K$-calculus; and an application of a labelled λ-calculus*, Theoretical Comput. Sci. 2 (1976) 97-114.

[29] C.-H. LUKE ONG, *The Lazy Lambda-Calculus: an Investigation into the Foundations of Functional Programming*, PhD Thesis, Dept. of Computing, Imperial College (1988).

[30] C.-H. LUKE ONG, *Fully abstract models of the lazy lambda calculus*, 29th FOCS (1988) 368-376.

[31] A.R. MEYER, *Semantical paradigms*, LICS 88 (1988) 236-253.

[32] R. MILNER, *Processes: a mathematical model of computing agents*, Logic Colloquium 73, North-Holland (1973) 157-173.

[33] R. MILNER, *Fully abstract models of typed λ-calculi*, Theoretical Comput. Sci. 4 (1977) 1-22.

[34] R. MILNER, *A modal characterisation of observable machine-behaviour*, CAAP 81, Lecture Notes in Comput. Sci. 112 (1981) 25-34.

[35] R. MILNER. J. PARROW, D. WALKER, *A calculus of mobile processes*, Technical Reports ECS-LFCS-89-85 & 86, LFCS, Edinburgh University (1989).

[36] R. MILNER, *Functions as processes*, INRIA Res. Report 1154 (1990).

[37] G. PLOTKIN, *A powerdomain construction*, SIAM J. Comput. 5 (1976) 452-487.

[38] G. PLOTKIN, *LCF considered as a programming language*, Theoretical Comput. Sci. 5 (1977)

223-256.

[39] G. PLOTKIN, T^{ω} *as a universal domain*, J. of Computer and System Sciences 17 (1978) 209-236.

[40] D. SCOTT, *Outline of a mathematical theory of computation*, 4^{th} Ann. Princeton Conf. on Information Sciences and Systems, Princeton University (1970) 169-176.

[41] D. SCOTT, *Data types as lattices*, SIAM J. Comput. 5 (1976) 522-587.

[42] D. SCOTT, *Lambda-calculus: some models, some philosophy*, The Kleene Symposium, North-Holland (1980) 223-265.

[43] D. SCOTT, *Domains for denotational semantics*, ICALP 82, Lecture Notes in Comput. Sci. 140 (1982) 577-613.

[44] A. STOUGHTON, *Substitution revisited*, Theoretical Comput. Sci. 59 (1988) 317-325.

[45] A. STOUGHTON, *Fully Abstract Models of Programming Languages*, Research Notes in Theoretical Computer Science, Pitman, Wiley (1988).

[46] B. THOMSEN, *A calculus of higher order communicating systems*, POPL 89 (1989) 143-154.

[47] C. WADSWORTH, *The relation between computational and denotational properties for Scott D_{∞}-models of the lambda-calculus*, SIAM J. Comput. 5 (1976) 488-521.

Imprimé en France
par
l'Institut National de Recherche en Informatique et en Automatique

ISSN 0249-6399